Modeling and discrimination for spectral-temporal data

Monica Marie Reising

Iowa State University
Modeling and discrimination for
spectral-temporal data

by

Monica Reising

A dissertation submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Statistics

Program of Study Committee:
Stephen Vardeman, Co-major Professor
Max Morris, Co-major Professor
Huaiqing Wu
Arka Ghosh
Frank Peters

Iowa State University
Ames, Iowa
2009

Copyright © Monica Reising, 2009. All rights reserved.
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS .............................................. v

ABSTRACT ........................................................... vi

GENERAL INTRODUCTION ........................................ 1

1 Motivation ......................................................... 1

2 Existing Methods of Discrimination .......................... 2

3 Chapter Introductions ........................................... 4

4 References ....................................................... 5

CHAPTER 1. MODELING SPECTRAL-TEMPORAL DATA FROM

POINT SOURCE EVENTS ........................................ 7

1 Abstract ............................................................. 7

2 Introduction ....................................................... 7

3 Data ............................................................... 9

4 Modeling ........................................................... 14

5 Estimating Model Parameters ................................. 16

5.1 Training Event Time Registration .............................. 17
5.2 Estimating the Class Mean and Standard Deviation Surfaces and Event-Specific “Vertical” Shift Parameters ........................................ 18
5.3 Estimating the Class Correlation Structure ........................................ 22
5.4 Modeling Summary ........................................................................ 37

6 Simulating and Assessing Data from Fitted Models ............................... 39

7 Conclusions ..................................................................................... 51

8 Acknowledgments ............................................................................ 51

9 References ....................................................................................... 61

CHAPTER 2. AN ALGORITHM FOR REAL-TIME DISCRIMINATION BASED ON SPECTRAL-TEMPORAL DATA FROM POINT SOURCE EVENTS ................................................................. 62

1 Abstract .......................................................................................... 62

2 Introduction ..................................................................................... 62

3 Modeling Background ........................................................................ 63

4 Development of a Discrimination Algorithm ....................................... 65

4.1 Determining Cut-Points for a Discrimination Algorithm .................. 67

4.2 Assessing the Performance of the Discrimination Algorithm ............ 70

4.3 Assessing the Performance of the Discrimination Algorithm on Real Data ............................................................................. 74

5 Discussion and Future Work ........................................................... 79

6 References ....................................................................................... 81
# CHAPTER 3. PRACTICAL CONSIDERATIONS FOR MODELING AND REAL-TIME DISCRIMINATION BASED ON SPECTRAL-TEMPORAL DATA FROM POINT SOURCE EVENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Abstract</td>
<td>82</td>
</tr>
<tr>
<td>2</td>
<td>Introduction</td>
<td>83</td>
</tr>
<tr>
<td>3</td>
<td>The Sensor</td>
<td>83</td>
</tr>
<tr>
<td>4</td>
<td>Raw Data Extraction</td>
<td>90</td>
</tr>
<tr>
<td>5</td>
<td>Intensity Shift Parameter</td>
<td>93</td>
</tr>
<tr>
<td>6</td>
<td>References</td>
<td>95</td>
</tr>
</tbody>
</table>

# CONCLUSIONS AND FUTURE WORK

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Conclusions</td>
<td>97</td>
</tr>
<tr>
<td>2</td>
<td>Future Research</td>
<td>97</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

I am truly grateful for the patience of my major professors, Stephen Vardeman and Max Morris. Both have contributed a great deal of time to me through regular meetings, reviewing manuscripts, offering advice, and listening. I am honored to have worked with not only great statisticians and professors, but truly amazing individuals.

I would also like to thank Alicia Carriquiry for her kind words and belief in my abilities. I am thankful for the opportunities of working with such an optimistic and enthusiastic individual as Alicia.

I would like to thank Huaiqing Wu, Arka Ghosh, and Frank Peters for taking time out of their busy schedules to act as committee members.

Finally, I thank my family and friends for all their support. Their patience and kind words were greatly appreciated as I completed my graduate work.
This thesis focuses on applying statistical methods to spectral-temporal data obtained from point source events. This work arises from the need in some military and national defense applications to quickly detect, locate, and identify short duration “energetic” electromagnetic events providing particular characteristic patterns of evolution over time. The first article discusses model building for spectral-temporal data that have complete spectral and temporal information over an event’s evolution. The goal of this work was to build models to serve as the basis for algorithms that can be used to distinguish between different types of electromagnetic events in real time.

The second article discusses the preliminary design of an algorithm for real-time discrimination between different types of point source events based on spectral-temporal data. The development of the algorithm was based on data obtained from 3 classes of safety matches and from simulated data based on fitted models developed in the first article.

The third and final article discusses important practical considerations regarding the sensor and experimental set-up used in the previous two articles. If this line of inquiry is to be further developed, this article discusses some practical considerations that should be addressed when moving forward.
GENERAL INTRODUCTION

1 Motivation

It is imperative in some military and national defense applications to quickly detect, locate, and identify short duration “energetic” electromagnetic events that have particular characteristic patterns of evolution over time. Spectral-temporal sensors (also known as pseudo-imagers) currently available capture information from energetic events and record observed intensities repeatedly in time for wavelengths ranging from the visible to the long-wave infrared (Pellegrini et al. 2004). Some of these important energetic events, particularly those associated with what are effectively point sources, can be difficult to detect, locate, and identify using traditional imaging cameras. Recently, considerable effort has been invested in pseudo-imaging sensors to improve their ability to detect and locate energetic events, using a direct vision prism to disperse incoming electromagnetic energy over a staring focal plane array (Deming et al. 2006).

While physical sensor technology is developing rapidly, there is a lag in the development of algorithms that can be used to identify and discriminate between types of energetic events in real time. There does not yet exist a coherent mathematical frame-
work for discrimination of energetic electromagnetic events that 1) explicitly allows for many sources of variability (shot-to-shot variability, atmospheric variation, sensor noise, etc.), 2) can be implemented in real-time, 3) cleanly handles “time registration of events” (recognizes that an incoming data stream may not contain data beginning at event initiation), and 4) can assess the likelihood that an incoming data stream matches the characteristic evolution of an “important” electromagnetic event.

2 Existing Methods of Discrimination

Several techniques have been investigated for discriminating energetic electromagnetic events. These include reducing the dimensionality of the data produced in both the mid-wave and near-infrared portions of the spectrum (wavelengths ranging from 0.7-20 microns). Orson et al. (2003) compared several data analysis techniques by applying them to data collected using a Fourier transform interferometer. Sensed intensity data were recorded over time at wavelengths from 1.6-20 microns on 56 detonation events for a variety of bomb sizes, types, and environmental conditions. Techniques investigated included analysis of fireball size as a function of time. The spectral profile was examined for several bomb types. Planckian distributions were fit to the data. This method was compared to fitting temporal overlap as a function of frequency. (The temporal overlap is defined as $O(\nu) = \cos(\theta) = \frac{\vec{S}_i(\nu) \cdot \vec{S}_j(\nu)}{|\vec{S}_i(\nu)||\vec{S}_j(\nu)|}$ where $\vec{S}_i(\nu)$ is a vector of radiance values at a given frequency $\nu$ for the $i^{th}$ event. Each time increment adds another element to the vector. The temporal overlap ranges from 0 (profiles are orthogonal) to 1 (profiles are identical).) The authors found that the temporal overlap classification method provided
a good signature for the classification of events. This method removes the dependence on total radiance and therefore reduces the variability of similar detonations. However, these methods are not based on the modeling of multiple sources of variability and so do not provide a firm basis for quantification of decision uncertainty. And while these methods have been used for classification of events, no attempt has been made to apply them in real time.

Gross et al. (2003) investigated the use of temporal profiles from data collected at wavelengths from 1.6-20 microns on two different static detonation events. In this approach, a nonlinear regression is fit to each of two parameterized functions for each time profile. The time profiles are constructed by averaging intensities across selected spectral bands. Then parametric forms are used to model the background signal and identify the start of the detonation, but again the dimensionality of the data is reduced. The authors feel this has opened the door to the classification of detonation events. However, they are unsure which parameters in the model are reproducible and which are key for classification. The methods discussed do allow classification of detonations, but not classification in real time.

Dills et al. (2004) also investigate a method for classification of battlespace detonations using data collected at wavelengths from 0.9-1.7 microns. In this approach features are extracted from the temporal profiles of different classes of data. An attempt is made to find features that represent differences between classes. Second order polynomials are fit to the data and the downward curvature is removed. The residuals are summarized by a probability density function where the minimum of the residuals
is set to zero. The first four moments of the probability distribution function are used as features. Using all the extracted features, Fisher ratios of all possible subsets (of size 3) of features are compared. The subset that provides that highest Fisher ratio provides the best discrimination. Next all possible combinations (of size 2) from those three features are investigated. It was found that discrimination was most effective using the first and second moments as the classifying features. Again, these methods are not being implemented in real time.

No attempt has yet been made to simultaneously model the characteristic patterns of energetic events for both time and wavelength or to make use of all the information produced in both time and wavelength without some form of (plausible, but nevertheless arbitrary) data reduction.

3 Chapter Introductions

We concentrate our efforts in Chapter 1 on building a mathematically coherent framework to effectively model the mean (characteristic signature) and covariance structures of energetic electromagnetic events evolving in time. A general framework will be developed that can, in theory, be applied to data streams produced by a sensor that detects intensities at any set of wavelengths in the spectrum and at any framing rate. The modeling framework described and applied in Chapter 1 will serve as the foundation upon which to build algorithms that can be used in real time to determine if an incoming data stream is consistent with that of a previously-modeled type of energetic event of interest.
The phenomenology of an energetic event will determine the optimal sensor and band pass (critical interval of wavelengths) to use for detecting such an event. Although tremendously important for designing a sensor, this is largely academic from a modeling point of view. Chapter 1 seeks to determine a framework for modeling that is applicable to nearly any pseudo-imageing sensor operating in any part of the electromagnetic spectrum that is relevant to its application.

Chapter 2 describes the development of one possible discrimination algorithm that could be used in real time. The algorithm is developed and properties are studied using the 3 classes of energetic events that were modeled in Chapter 1. Chapter 2 describes the number of false rejections for each of 3 classes of safety matches and the rate to rejection.

Finally, Chapter 3 describes components of the sensor that was used in data collection. Additionally, Chapter 3 discusses the raw data extraction and some practical considerations to implementing the discrimination algorithm in real time.

4 References


ceedings of SPIE - The International Society for Optical Engineering. 5431-10, pp 77-86.


1 Abstract

In recent years, a great deal of effort has been invested in developing sensors to detect, locate, and identify “energetic” electromagnetic events. When observed through one type of imaging spectrometer, these events produce a data record that contains complete spectral and temporal information over the event’s evolution. This article discusses model building for spectral-temporal data of this type. The goal is to build models that can be used in future work to produce algorithms that can be applied to sensor data to distinguish between different types of electromagnetic events in real time.

2 Introduction

It is imperative in some military and national defense applications to quickly detect, locate, and identify short duration “energetic” electromagnetic events that have par-
ticular characteristic patterns of evolution over time. Spectral-temporal sensors (also known as pseudo-imagers) currently available capture information from energetic events and record observed intensities repeatedly in time for wavelengths ranging from the visible to the long-wave infrared (Pellegrini and Ewing 2004). Some of these important energetic events, particularly those associated with what are effectively point sources, can be difficult to detect, locate, and identify using traditional imaging cameras. Recently, considerable effort has been invested in pseudo-imaging sensors that use a direct vision prism to disperse incoming electromagnetic energy over a staring focal plane array (Deming et al. (2006)).

While physical sensor technology is developing rapidly, there is a lag in the development of algorithms that can be used to identify and discriminate between types of energetic events in real time. There does not yet exist a coherent mathematical framework for discrimination of energetic electromagnetic events that 1) explicitly allows for many sources of variability (shot-to-shot variability, atmospheric variation, sensor noise, etc.), 2) can be implemented in real time, 3) cleanly handles “time registration of events” (recognizes that an incoming data stream may not contain data beginning at event initiation), and 4) can assess the likelihood that an incoming data stream matches the characteristic evolution of an “important” electromagnetic event.

While developing detection algorithms is the ultimate goal, we concentrate our effort here on building a mathematically coherent framework to effectively model the mean (characteristic signature) and covariance structures of energetic electromagnetic events evolving in time. A framework will be developed that could be applied to data streams
produced by a sensor that detects intensities in any set of wavelengths and at any framing rate. The framework described in this article serves as the foundation for subsequent work in which we build algorithms that can be used in real time to determine if an incoming data stream is consistent with that of a previously-modeled type of energetic event.

The phenomenology of an energetic event of interest will determine the optimal sensor and band pass (set of wavelengths to which it is sensitive) to use for detecting it. Although important for designing a sensor, such considerations are not of basic concern from a modeling point of view. This paper seeks to develop a framework for modeling that is broadly applicable to pseudo-imaging sensors operating in any part of the electromagnetic spectrum that is of interest. Our concentration on the infrared part of the spectrum in our case study is thus not intrinsic to our analysis.

3 Data

Our test bed for developing statistical models consists of data collected by a slightly modified sensor on burns of three types of safety matches. Safety matches (as opposed to other potential hot sources of radiation) were used so that we could easily control experimental conditions, and data for many event repetitions were collected.

The sensor we used typically employs a technique known as pseudo-imaging (or imaging) to provide spectral-temporal signatures and the locations of rapidly changing events within a given field of view. Pseudo-imaging is the process of taking data obtained from the focal plane array, creating a single spectral profile, and using the spectral profiles
from multiple ‘snap-shots’ to create a spectral-temporal signature from an observed event (Weeks et al. (2004)). Figure 1 shows a schematic of the sensor we used from Mooney, Vickers, An, and Brodzik (1997) and Pellegrini and Ewing (2004).

Data were collected in May 2005 at Hanscom Air Force Base, MA. Three different types of safety matches were used to create realizations from three different classes of energetic events. Data were recorded for 20 matches of each of type. The sensor used was set at a framing rate of 80 frames per second. After the sensor began recording, a match was ignited by hand and the sensor continued to collect data for nine seconds, producing 720 frames of measured intensities. (A total of 720 frames were recorded for each test regardless of whether the match burned the entire time or not.) For each of the 720 frames of data, a gray-scale image was produced. After all data were collected, the collection of pixels containing the brightest line in each gray-scale image was manually
extracted. The intensity (read out voltages as a function of photon counts) recorded for each pixel along the line were compiled to create a matrix of data; each row in this matrix corresponds to a single temporal frame, and each column to a single pixel on the focal plane array (an index for convenience corresponding to a wavelength in the spectrum). These are used to make what is commonly referred to as a “waterfall plot.” Figure 2 is such a plot.

In normal use, the sensor is equipped with a rotating prism and rotating preprocessing algorithm that can produce waterfall plots like that in Figure 2. However, for the purpose of initial development of modeling methods, we decided to not rotate the prism during data collection and began with (not preprocessed to account for rotation, but rather) raw data from a single column of pixels on the focal plane array.

As mentioned previously, data were recorded for 20 matches of three types, say Class A, Class B, and Class C. In the process of collecting data for Class C, the prism was realigned after the first three observations. While preprocessing the data, we discovered that data for those three observations were extracted from a different column of pixels on the focal plane than the others. After careful consideration, these observations were not used in the development of the model for Class C.

Class A and Class B models are expected to be similar, as a match from Class A was simply two matches from Class B physically bound together. The chemical compositions of the Class A and Class B matches were identical. The matches from Class C are different in terms of chemical composition and the Class C model should be substantially different from those for Classes A and B.
Figure 2  An Example of Raw Data Collected in the Burn of a Single Match from Class A

Let the measured energy intensity observed by a sensor for one event be denoted by

$$S^*(t,l)$$ for $$t = 0, 1, \ldots$$ and $$l = 0, 1, \ldots$$ \hspace{1cm} (1)

where $$l$$ is an integer index used to represent a pixel on the focal plane array. $$S^*_k(t,l)$$ will be used to represent the measured energy at the $$t^{th}$$ time frame on the $$l^{th}$$ pixel for the $$k^{th}$$ observation for a fixed class of matches. In the raw data sets, $$t = 0, \ldots, 719$$ and $$l = 1, \ldots, 150$$, since the portion of the gray-scale images used in our study consists of 150 consecutive pixels in a column of the focal plane array. Each value of $$l$$ corresponds to a single pixel on the focal plane (since the spectral resolution of the data is a single pixel) and integer indexing was used for convenience. Note that a given value of $$l$$, say 80, corresponds to the same pixel on the focal plane for all time frames, and that any
two consecutive values of \( l \) correspond to neighboring pixels on the focal plane. After preliminary examination of the raw data, we determined some additional preprocessing would be needed (a “background subtraction”). Upon completion of the preprocessing, we had “useable” data like those portrayed in Figure 3.

![Figure 3](image)

Figure 3  Background-Subtracted Data (Corresponding to Those Represented in Figure 2)

Let

\[
S(t, l) \text{ for } t = 0, 1, \ldots, 719 \text{ and } l = 1, \ldots, 149
\]

(2)

represent preprocessed data for a single event. Here a value of \( l \) corresponds to the same pixel on the focal plane array for all event classes.
4 Modeling

To begin model development, suppose that a bright event of a particular type produces a sensed spectrum that evolves over continuous time ($\tau$) and wavelength ($\lambda$) as

$$\theta(\tau, \lambda) \text{ for } \tau \in (0, T) \text{ and } \lambda \in (0, \infty).$$ (3)

We purposely use the notation $\theta(\tau, \lambda)$ here rather than $S(t, l)$, because for modeling purposes we wish to think of $\tau$ and $\lambda$ as continuous variables and to do the modeling for all $\tau \in (0, T)$ and $\lambda$ in some finite interval in $(0, \infty)$. The rationale here is that a point source has a signature that is continuous in time for some interval of wavelengths.

A model is needed for $\theta(\tau, \lambda)$ that explicitly allows for shot-to-shot (realization-to-realization) variability. We will use a Gaussian random field model for a transformed version of $\theta(\tau, \lambda)$ which we will call $\tilde{\theta}(\tau, \lambda)$. The exact transformation used in a particular case will be dictated by the data. (It is possible that each class of events might require a different transformation.)

Because physical intensities are non-negative (read-out voltages as a function of photon counts) it is natural in this application to think of using a logarithmic transformation. A log transformation would convert any multiplicative effects (such as those produced by distance or atmospheric absorption) on sensed signals to additive effects and mitigate the “orders of magnitude” differences between responses seen across $t$ and $l$ pairs. However, due to the background subtraction and instrument noise, the preprocessed data (2) can include negative values. A transformation was needed that would have characteristics of the log transform where there is substantial signal, but that is applicable to
both positive and negative data values. An invertible transformation is also desirable so as to avoid information loss. Substantial informal preliminary analysis suggested that the monotone transformation

\[
h(\theta) = \begin{cases} 
\ln(\theta) & \text{for } \theta \geq 2 \\
\theta \left(\ln(2)/2\right) & \text{for } -2 < \theta < 2 \\
-\ln(|\theta|) & \text{for } \theta \leq -2 
\end{cases}
\] (4)

is useful for all classes in our test bed data. Figure 4 shows the application of transform (4) to the values represented in Figure 3.

Figure 4  Transformed Data Corresponding to Those Represented in Figures 2 and 3
A Gaussian random field model for

$$\tilde{\theta}(\tau, \lambda) = h(\theta(\tau, \lambda))$$

is characterized by a mean function $\mu(\tau, \lambda)$ and a covariance function $C((\tau_1, \lambda_1), (\tau_2, \lambda_2))$ defined for $\tau, \tau_1$ and $\tau_2 \in (0, T)$ and $\lambda, \lambda_1$ and $\lambda_2$ belonging to the interval of interest in $(0, \infty)$. For any finite set of time-wavelength pairs $(\tau_i, \lambda_i), i = 1, 2, \ldots, m$, the joint distribution of values $\tilde{\theta}(\tau_i, \lambda_i)$, is multivariate normal with mean vector

$$E(\tilde{\theta}(\tau_i, \lambda_i)) = \mu(\tau_i, \lambda_i)$$

and covariance matrix with entries

$$\text{Cov}(\tilde{\theta}(\tau_i, \lambda_i), \tilde{\theta}(\tau_j, \lambda_j)) = C((\tau_i, \lambda_i), (\tau_j, \lambda_j)).$$

5 Estimating Model Parameters

Recall that the preprocessed data are

$$S(t, l) \text{ for } t = 0, 1, \ldots, 719 \text{ and } l = 1, 2, \ldots, 149$$

as in (2), to be transformed as

$$\tilde{S}(t, l) = h(S(t, l))$$

before modeling.

After substantial exploratory data analysis, we found that a plausible model for the $k^{th}$ observation within a given class has, corresponding to (9), a decomposition of $\tilde{\theta}(\tau, \lambda)$ as

$$\tilde{\theta}_k(\tau, \lambda) = d_k + \mu(\tau + \eta_k \Delta \tau, \lambda) + \sigma(\tau + \eta_k \Delta \tau, \lambda)\epsilon(\tau, \lambda).$$
In (10) the index \( k \) identifies a specific realization from the class, and \( d_k \) (intensity shift) and \( \eta_k \) (time shift) are parameters associated with the \( k^{th} \) event and \( \Delta \tau \) is the time difference between two consecutive sampling points. \( \mu \) and \( \sigma \) are mean and standard deviation functions associated with the particular class of events and \( \epsilon \) is a “spatial” stochastic process with mean zero, variance one, and a correlation structure potentially specific to the class. We proceed to describe how we developed appropriate forms for \( \mu(\cdot), \sigma(\cdot), \) and the correlation structure for \( \epsilon(\cdot, \cdot). \)

5.1 Training Event Time Registration

The raw data consist of 720 frames for each observation from each class of events, but the physical bright event did not start in the same frame of data for each observation. To estimate appropriate mean and standard deviation functions for characterizing a class, the bright events need to be aligned temporally. This alignment is accomplished by selecting values \( \eta_k \) in equation (10). The estimates of \( \eta_k \) for observations in a class were calculated from total intensity series. Let

\[
I_k(t) = \sum_l S_k(t, l)
\]

(11)

denote the total intensity at time \( t \) for event \( k \) from the class under consideration. The temporal alignment parameter, \( \eta_k \), was estimated as

\[
\hat{\eta}_k = \min_k (\arg \max_t I_k(t)) - \arg \max_t I_k(t).
\]

(12)

That is, the realization with the earliest intensity “peak” was selected as a “base case,” and time-offsets for all other realizations are determined with reference to it. Figure 5
illustrates a subset of total intensity series for one class of events and Figure 6 shows the same subset of total intensity series aligned according to the values of the $\hat{\eta}_k$ given in (12).

![Total Intensity Series](image)

Figure 5 Total Intensity Series Prior to Temporal Alignment

5.2 Estimating the Class Mean and Standard Deviation Surfaces and Event-Specific “Vertical” Shift Parameters

It is well-known that the distance from which bright events are observed will affect the intensities registered by the sensor. If two identical events are observed by the camera through the same atmosphere at two different distances, the event closer to the sensor registers higher intensity values than the event that is observed from a greater distance. This phenomenon introduces the need for $d_k$ in model (10). This value is an
intercept, or an “intensity shift” parameter. To estimate the intensity shift parameter, $d_k$, for an observation, the mean function for the class under consideration, $\mu(\tau, \lambda)$, must first be estimated. For every $(t, l)$ pair (after temporal alignment) an average was taken across the $k$ observations within a class. The domain of the empirical mean function was trimmed so that only those intensities corresponding to times where all observations make a contribution to the average are included. (That is, we effectively reindex time for all realizations to match these with $\hat{\eta}_k = 0$ and only estimate mean response for such $t = 0, 1, \ldots, 719 - \max_k|\hat{\eta}_k|$ and $l = 1, 2, \ldots, 149$.) Let the estimated mean function be denoted by $\mu'(t, l)$ for such $t$ and $l$.

With an estimated mean function in hand, we estimated an appropriate “intensity
shift, $d_k$, for each observation. We computed this as

$$
\hat{d}_k = \frac{1}{149(719 - \max_k |\hat{\eta}_k|)} \sum_{t=0}^{719-\max_k |\hat{\eta}_k|} \sum_{l=1}^{149} (\tilde{S}_k(t + \hat{\eta}_k, l) - \mu'(t, l)).
$$

(13)

After the mean function and intensity shift parameters are estimated, residuals may be computed as

$$
\hat{e}_k(t, l) = \tilde{S}_k(t + \hat{\eta}_k, l) - (\hat{d}_k + \mu'(t, l)).
$$

(14)

Note that for each $(t, l)$ pair, the average of these residuals over all observations is zero. Then the class-specific variance at each observed time-wavelength pair $(t, l)$ was estimated as

$$
\tilde{\sigma}^2(t, l) = \frac{1}{N - 1} \sum_k (\hat{e}_k(t, l))^2.
$$

(15)

where $N$ is the number of replicate trials from the class. Final estimates of the mean and variance functions are computed as smoothed versions of $\mu'(t, l)$ and $\tilde{\sigma}^2(t, l)$. A loess smoothing technique is used with a smoothing parameter of 0.1. (With a smoothing parameter of 0.1, the smoother uses the nearest ten percent of observations and replaces the current observation with a tricubic weighted average of those nearest observations.)

Figures A.1-A.3 in the Appendix show the smoothed mean and standard deviation surfaces. (The mean surface is displayed to the left of the standard deviation surface for each class.) The first 35 wavelength values (the left side wavelengths) were removed prior to plotting. This was done so the surface could be “seen” for the middle wavelengths. These middle wavelengths are where the standard deviation surfaces vary the most from event class to event class.

Next, standardized residuals may be calculated using $\hat{\mu}(t, l)$ and $\hat{\sigma}(t, l)$. For each
observed time-wavelength pair \((t, l)\) the standardized residuals for the \(k^{th}\) observation are

\[
\hat{\xi}_k(t, l) = \frac{\tilde{S}_k(t + \hat{\eta}_k, l) - (\hat{d}_k + \hat{\mu}(t, l))}{\hat{\sigma}(t, l)}.
\]  

(16)

\(\hat{\xi}_k(t, l)\) is constructed to have approximately mean zero and standard deviation one for each \((t, l)\) pair (across the events in the class). These are treated as partial realizations of the stochastic process \(\epsilon(\tau, \lambda)\) in Equation (10), and used in estimating the covariance function for that process. Figure 7 is a plot of the standardized residuals \(\hat{\xi}(t, l)\) for the raw data represented in Figure 2.

Figure 7  A Single Standardized Residual Surface (Corresponding to the Data Represented in Figure 2)
5.3 Estimating the Class Correlation Structure

Some exploratory data analysis suggested that it is adequate (for each event type) to segment the set of observed \((t, l)\) pairs into four (rectangular) sets by what we will term (class-specific) “independence walls.” The corresponding four sets of \(\epsilon(\tau, \lambda)’s\) are assumed to be independent of each other. Figure 8 gives a diagram of the independence walls. The left side corresponds to shorter wavelengths (small values of \(l\) in our indexing) and the right side corresponds to longer wavelengths. The vertical line represents an independence wall built into the model separating \((t, l)\) pairs at the undeviated wavelength (the wavelength of energy that passes through the prisms undeflected). Our modeling assumption is that any \(\epsilon(\tau, \lambda)\) with \(\lambda\) below the undeviated wavelength is independent.

![Independence Walls in \((t, l)\) Space Showing 4 Regions](image)
of any $\epsilon(\tau, \lambda)$ with $\lambda$ above the undeviated wavelength. The horizontal line represents an independence wall built into the model in the time direction. This separates what we will call the event initiation phase (or early time) from what we will call the steady-state phase (or late time). In Figure 6, the steady-state phase is represented by the part of the plot following the sharp peak and drop in total intensity where the total intensity series tends to vary relatively slowly. One can now think of the data as separated into four sets defined by the regions seen in Figure 8, right late time region, right early time region, left late time region, and left early time region.

Furthermore, due to patterns remaining in the apparent “local roughness” of the standardized residual surfaces (see Figure 7), correlation structures were investigated separately in sub-regions within the four regions (see Figure 9). “Local roughness” in realizations of a stochastic process is indicative of the nature of the correlation structure, smoothness indicating high correlation between responses with $(\tau, \lambda)$ in close proximity to each other. In the regions to the left of the undeviated wavelength, in both early and late time, $(t, l)$ pairs were further divided into three sub-regions in the wavelength direction that we will call the “up” sub-region, the “middle” sub-region, and the “down” sub-region according to the behavior of the estimated mean function $\hat{\mu}(t, l)$ in them. On the right side in the wavelength direction, in both early and late time, $(t, l)$ pairs were further divided into four sub-regions, the “up” sub-region, the “top” sub-region, the “middle” sub-region, and the “down” sub-region according to the behavior of the estimated mean function in these regions. These sub-regions do not correspond to equal numbers of wavelength indices and all of the partitioning indicated in Figure 9 was done
based on the empirical patterns seen in the training data for the three classes. So, for example, the sizes of the sub-regions vary from class to class in order to effectively model the different types of “burns” seen in the data. The goal in dividing the \((t, l)\) plane into sub-regions in the observed wavelength direction was to allow us to change the scales on the wavelength axis, sub-region to sub-region, so that a single stationary correlation function could be used across each region (after transforming wavelength). This limited the number of parameters needed to specify correlation structures. (It is worth noting that this change of scale is reversible and there is no loss of information involved.)

![Regions Further Divided Into Sub-Regions](image)

Figure 9  Sub-Regions Within the Four \((t, l)\) Regions

In seeking a simple correlation structure for \(\epsilon(\tau, \lambda)\), our hope was that a product form (in \(\tau\) and \(\lambda\)) would be adequate. Accordingly, our attention turned to calculating
variograms of the standardized residuals, \( \hat{\xi}_k(t, l) \), for each class of objects in the time and wavelength directions separately. The use of variograms is common in spatial statistics applications when estimating functional forms for covariance structures. A natural estimator (based on the method of moments) for the variogram was calculated in the wavelength direction (for each event and fixed time). (See Cressie (1993) beginning on page 58 for a discussion of variograms and their estimation.) Likewise, for each event and fixed wavelength, the empirical variogram was calculated in the time direction.

When seeking a simple functional form to fit to the empirical variograms in the wavelength direction for fixed time, we noticed that the “nugget” appeared to decrease in time in our test bed data. (The nugget is the size of the apparent discontinuity of a variogram at the origin, representing the scale of the uncorrelated “white noise” portion of variation in a stochastic process.) After further careful exploratory analysis, we decided to adopt a modeling strategy that partitions the stochastic process into a white noise component for which the variance is a function of time and wavelength, independent of a second (correlated) component modeled using a product covariance structure. That is, we propose to employ a decomposition of the stochastic process as

\[
\epsilon(\tau, \lambda) = \epsilon_1(\tau, \lambda) + \epsilon_2(\tau, \lambda),
\]

where,

\[
E(\epsilon_1(\tau, \lambda)) = 0 \quad \forall \tau, \lambda \quad \text{and} \quad \text{Var}(\epsilon_1(\tau, \lambda)) = \delta^2(t, \lambda) \quad \forall t, \lambda,
\]

\[
\text{Cov}(\epsilon_1(\tau, \lambda), \epsilon_1(\tau', \lambda')) = 0 \quad \text{for} \quad \tau \neq \tau' \quad \text{or} \quad \lambda \neq \lambda',
\]

\[
E(\epsilon_2(\tau, \lambda)) = 0 \quad \forall \tau, \lambda \quad \text{and} \quad \text{Var}(\epsilon_2(\tau, \lambda)) = 1 - \delta^2(\tau, \lambda) \quad \forall \tau, \lambda,
\]
\[ \text{Cov}(\epsilon_2(\tau, \lambda), \epsilon_2(\tau', \lambda')) = R_{\tau}(|\tau - \tau'|)R_{\lambda}(|\lambda - \lambda'|) \times \sqrt{(1 - \delta^2(\tau, \lambda))(1 - \delta^2(\tau', \lambda'))} \quad \forall \, \tau, \lambda \text{ and } \tau', \lambda' \text{ and,} \]

\[ \text{Cov}(\epsilon_1(\tau, \lambda), \epsilon_2(\tau', \lambda')) = 0 \quad \forall \, \tau, \lambda \text{ and } t', \lambda', \]

where \( R_{\tau}(|\tau - \tau'|) \) represents a nuggetless (i.e continuous at the origin) correlation function in the time direction and \( R_{\lambda}(|\lambda - \lambda'|) \) represents the nuggetless correlation function in the wavelength direction. We emphasize that \( R_{\lambda} \) is not assumed to be the same function in each sub-region and that our model will eventually allow for non-zero correlation between standardized errors from sub-regions within the same region.

In order to begin fitting the decomposition described in (17) to the standardized residuals in a given region, we first needed to estimate \( \delta^2(\tau, \lambda) \). This cannot be done directly because there are no “true replicates” in the data, but we can obtain an estimate based on the degree of local variability near each \((t, l)\). To this end, let

\[ Y_k(t, l) = \hat{\xi}_k(t, l) - \frac{1}{4}[\hat{\xi}_k(t, l - 1) + \hat{\xi}_k(t, l + 1) + \hat{\xi}_k(t - 1, l) + \hat{\xi}_k(t + 1, l)]. \quad (18) \]

\( Y \) is, for each observed standardized residual, a difference between \( \hat{\xi} \) and the average of its four nearest neighbors in time and wavelength. Intuition for this quantity was that it should roughly mimic \( \epsilon_1 \).

\( Y_k^2(t, l) \) was calculated for each object in a class and each observed \((t, l)\) combination, and these values were averaged across all observations within a class, for each \((t, l)\) pair. The average was then smoothed using a loess smoothing technique. Calculation of \( Y^2 \) from equation (18) obviously does not apply to edge points in a region, since they do not have four nearest neighbors (as defined in equation (18)) in the original dataset. After smoothing \( Y^2 \) values not located on a \( t- \) or \( l \)-edge, each sequence of values for a fixed
time was examined. A functional form was fit for the small and large $l$ values (for each fixed time point). A non-negative $Y_k^2$ value for the first and last $l$ value for each fixed time step was extrapolated. A similar procedure was used to extrapolate $Y_k^2$ for the first and last time step (for each fixed $l$). Let $\tilde{Y}_k$ denote smoothed $Y_k$ with the extrapolated edge values.

Notice that the expectation of $Y_k(t, l)$ in (18) is zero for all $t, l$. Hence,

$$E(Y_k^2(t, l)) = \text{Var}(Y_k(t, l)).$$

Therefore, it can be shown (see the Appendix)

$$\text{Var}(Y_k(t, l)) \approx \frac{5}{4} \delta_k^2(\tau, \lambda).$$

Hence, the white noise variance function might be estimated for a particular region for a particular event class as

$$\hat{\delta}_k^2(t, l) = \frac{4}{5} \tilde{Y}_k^2(t, l) \quad \forall \ t, \ l.$$  (21)

Figures A.4-A.6 show estimated $\hat{\delta}_k^2$ surfaces for all four regions for each of the three classes of events.

With estimates (21) in hand, we turned our attention to estimating parametric forms for the nuggetless correlation structure (the correlation structure for $\epsilon_2$ in equation (17)) in both the time and the wavelength directions. We based these estimates on the relationships

$$E(\epsilon_k(\tau, \lambda) - \epsilon_k(\tau, \lambda'))^2 = 2 - 2\sqrt{(1 - \delta_k^2(\tau, \lambda))(1 - \delta_k^2(\tau, \lambda'))}R_\lambda(|\lambda - \lambda'|).$$  (22)
and

$$E(\epsilon_k(\tau, \lambda) - \epsilon_k(\tau', \lambda))^2 = 2 - 2\sqrt{(1 - \delta_k^2(\tau, \lambda))(1 - \delta_k^2(\tau', \lambda))} R_\tau(|\tau - \tau'|).$$ (23)

To estimate a sub-region correlation function in the wavelength direction specific to the data here, $R_\ell(|l - l'|)$, first, for each fixed time index $t$ and pair of wavelength indices $l'$ and $l''$, we computed

$$X_{L,k}(t, l', l'') = \frac{2 - (\hat{\xi}_k(t, l') - \hat{\xi}_k(t, l''))^2}{2\sqrt{(1 - \delta_k^2(t, l'))(1 - \delta_k^2(t, l''))}}.$$ (24)

Similarly, to estimate the correlation function in the time direction specific to the data here, $R_t(|t - t'|)$, first, for each fixed wavelength index $l$ and pair of time indices $t'$ and $t''$, we computed

$$X_{T,k}(t', t'', l) = \frac{2 - (\hat{\xi}_k(t', l) - \hat{\xi}_k(t'', l))^2}{2\sqrt{(1 - \delta_k^2(t', l))(1 - \delta_k^2(t'', l))}}.$$ (25)

For each fixed time index $t$, values $X_{L,k}(t, l', l'')$ were then averaged according to

$$\hat{X}_L(t, \Delta l) = \frac{\sum N(\Delta l)(X_{L,k}(t, l', l''))}{|N(\Delta l)|},$$ (26)

where $N(\Delta l) \equiv \{(l', l'') : |l' - l''| = \Delta l\}$ and $|N(\Delta l)|$ is the number of distinct pairs in $N(\Delta l)$. A similar averaging was done with the $X_{T,k}(t', t'', l)$ for each fixed wavelength index $l$.

The statistics used to estimate the correlation functions were sequences of the above-calculated values, specifically

$$\hat{R}_L(t) = (\hat{X}_L(t, 1), \hat{X}_L(t, 2), \ldots, \hat{X}_L(t, \max \Delta l))$$ and
\[ \hat{R}_T(l) = (\hat{X}_T(1, l), \hat{X}_T(2, l), \ldots, \hat{X}_T(\max \Delta t, l)). \]

To find a single parametric form for a correlation function in the wavelength direction, we averaged all of the values \( \hat{R}_L(t) \) across the appropriate \( t \) in a given observation. These empirical variograms in the wavelength direction were constructed separately for each sub-region. Plots of these averages versus lag in wavelength index, \( \Delta l \), were examined and a parametric form was chosen. Figure 10 shows plots of \( \hat{R}_L(t) \) averaged across time values for the four sub-regions within a region for a single observation from one class of events. In the wavelength direction, these plots were similar from observation to observation within a class and suggested that a reasonable single form for a correlation function might be Gaussian, i.e. of the form \( \exp(-\beta (\Delta l)^2) \). The Gaussian correlation structure is associated with smooth realizations, which is consistent with the physical nature of our data.

\( \hat{R}_L(t) \) was computed for each \( t \) for each sub-region within each region for each observation and then averaged across time to produce \( \bar{R}_L \) for each observation in the training data (for the given sub-region for the relevant event class). Nonlinear Least Squares was used to estimate the parameters of the Gaussian form expanded to include a nugget term,

\[ \beta_1 + \exp(-\beta_2 (\Delta l)^2). \]

The values of \( \beta_1 \) in the final forms used will be set equal to zero. By our assumptions, the correlation function must approach a value of 1 as \( \Delta l \) approaches zero (i.e. \( \beta_1 = 0 \)). However, in obtaining a good estimate of correlations at short lags, this fitted form was used to avoid biasing estimates of \( \beta_2 \) when there is some lack-of-fit at small wavelength
Figure 10  Empirical Correlations $\hat{X}_L$ at Various Wavelength Lags Averaged Across Time for all Sub-regions Within a Single Region for One Event Class (Corresponding to the Data Represented in Figure 2)

differences. After the fitting, the empirical distributions of fitted $\beta_1$ and $\beta_2$ for a given sub-region and event class across the observations for that class were examined. The distributions of observation-specific $\beta_1$ estimates were centered around zero, as one would expect. A final single estimate of $\beta_2$ used in the correlation function for modeling in a given sub-region within a given region for a specific class of events was taken to be the median of the empirical distribution of observation-specific $\beta_2$ estimates; we denote this single value as $\hat{\beta}_2$.

The $\hat{\beta}_2$ values for the various sub-regions varied widely within a given region, indicating that even within a region a correlation structure appropriate to modeling the $\epsilon_2$ of
equation (17) cannot be stationary in $l$ across an entire region. Nevertheless, by changing scale for wavelength indices (or introducing a “pseudo-wavelength” corresponding to each original wavelength index) it is possible to make use of a single (stationary) Gaussian correlation structure in wavelength (thereby assuring that what we have specified is indeed mathematically coherent/valid).

To illustrate, in the early-time-left-side region, there were estimated values for $\beta_2$ for each sub-region, denoted as $\hat{\beta}_{2(up)}$, $\hat{\beta}_{2(middle)}$, and $\hat{\beta}_{2(down)}$. Using these values, and setting $\hat{\beta}_{2(up)}$ as a reference value, the size of the pseudo-distance (between adjacent wavelength indices) in a particular sub-region of the region will be denoted as $\tilde{l}^*$

$$\tilde{l}^* = \sqrt{\frac{\hat{\beta}_{2(up)}}{\hat{\beta}_{2(\cdot)}}}. \quad (28)$$

$\tilde{l}^*$ represents the “distance” between wavelength indices in a given sub-region relative to a “distance” of 1 in the reference sub-region. To obtain an appropriate vector of pseudo-distances for a given sub-region, the pseudo-wavelengths needed to be spaced appropriately. The (original and pseudo-) wavelength indices in the reference sub-region, $(l_{1(up)}, l_{2(up)}, \ldots, l_{last(up)})$ are spaced so that consecutive values differ by 1, i.e. $|l_i - l_j| = |i - j|$ $\forall$ $i, j$. The pseudo-wavelengths in the middle sub-region are then defined as

$$\tilde{l}_{n(middle)} = l_{last(up)} + n \times \tilde{l}^*_{(middle)} \text{ for } n = 1, 2, \ldots, N_{(middle)} \quad (29)$$

where $N_{(middle)}$ is the number of wavelength indices in the middle sub-region. Similarly, the pseudo-wavelengths for the down sub-region are defined as

$$\tilde{l}_{n(down)} = l_{last(middle)} + n \times \tilde{l}^*_{(down)} \text{ for } n = 1, 2, \ldots, N_{(down)}. \quad (30)$$
For example, suppose we have $\hat{\beta}_2^{(up)} = 0.02$, $\hat{\beta}_2^{(middle)} = 0.0002$ and $\hat{\beta}_2^{(down)} = 0.0242$ and wavelength indices in the three sub-regions: (1, 2, 3, 4), (5, 6, 7, 8, 9, 10), and (11, 12, 13). Then $\tilde{l}^{*\,(middle)} = 10$ and $\tilde{l}^{*\,(down)} = 0.9$ and original wavelength indices ($l$’s) map into pseudo-wavelengths ($\tilde{l}$’s) (1, 2, 3, 4), (14, 24, 34, 44, 54, 64), and (64.9, 65.8, 66.7).

Based on these pseudo-wavelengths a single stationary (in pseudo-wavelength) correlation function for the $\epsilon_2$ process in the wavelength direction within each class and region was used. That is, for a given region, the correlation function for the $\epsilon_2$ process in the wavelength direction is

$$ \exp(-\hat{\beta}_2^{(up)}(\tilde{l} - \tilde{l}')^2) $$

for $\tilde{l}'$ and $\tilde{l}''$ pseudo-wavelengths corresponding to wavelength indices $l'$ and $l''$. A similar process was used to estimate a single correlation function in the right side regions.

In a development parallel to the foregoing, we modeled correlations in the $\epsilon_2$ process in the time direction. First, $\tilde{R}_T(l)$ was computed for each $l$ in both the early and late time regions for each observation and then averaged across all wavelength indices to produce $\bar{R}_T$ for each observation in the training data (for early time and late time for each observation in the relevant event class). A parametric form was then fit to these averaged series.

Figure 11 shows $\bar{R}_T$ plotted against $\Delta t$ for four observations early in time from one class. It is apparent that there was a great deal of inconsistency in the form of estimated time correlations of observations. The non-monotone patterns seen in some plots are not easily explained except as estimation noise and sensor artifact. Ultimately, we decided to represent time correlation with a simple linear function.
Figure 11  Empirical Time Correlations for Four Observations Within A Class

The $\bar{R}_T$ series were truncated at $\Delta t = \min_k \arg\min_{\Delta t} \bar{R}_{T,k}(\Delta t)$ (producing series of the same length for different observations indexed by $k$). Least Median Squares was used (one observation at a time) to estimate the parameters of the linear form

$$\bar{R}_T(\Delta t) \approx \alpha_1 + \alpha_2 \Delta t.$$  \hspace{1cm} (32)

(Again, our assumption is that there is no nugget in this from, so the value of $\alpha_1$ in the final form will be set equal to one.) The median of the fitted values of $\alpha_2$ for a class, denoted by $\hat{\alpha}_2$, was used as the estimate for the entire region. Ultimately (so that the correlation at time lag 0 is 1 and no negative correlations are produced) we specify that for a given region, the correlation for the $\epsilon_2$ process in the time direction for our data was (for $\hat{\alpha}_2$ specific to the class of events and region) $\max(0, 1 + \hat{\alpha}_2|t' - t''|)$. 
Having developed plausible wavelength and time correlation models for $\epsilon_2$ for each region for each event class, it remained to combine the wavelength and time functions to create a single correlation function for a given region. One simple and mathematically coherent way to do this was to adopt a product correlation form. That is, if $\beta_2(\text{reference})$ and $\alpha_2$ are the wavelength and time correlation parameters appropriate for a given region for a particular event class, then we adopted a model that specifies that the correlation for the $\epsilon_2$ process for our data was, for time indices $t'$ and $t''$ and wavelength indices $l'$ and $l''$,

$$\text{Corr}(\epsilon_2(t', l'), \epsilon_2(t'', l'')) = \exp(-\beta_2(\text{reference})(\tilde{l}' - \tilde{l}'')^2)\max(0, 1 + \alpha_2|t' - t''|).$$ (33)

where $\tilde{l}'$ and $\tilde{l}''$ are pseudo-wavelength indices corresponding to $l'$ and $l''$, respectively.

We were now ready to assemble a single covariance structure for $\epsilon(\tau, \lambda)$ of displays (10) and (17) (the standardized error process) and finally a covariance structure for $\tilde{\theta}(\tau, \lambda)$. This is most economically described in matrix terms.

For a given region with time indices $t = t_{\min}, t_{\min}+1, \ldots, t_{\max}$ and wavelength indices $l = l_{\min}, l_{\min}+1, \ldots, l_{\max}$ there are $(t_{\max} - t_{\min} + 1)(l_{\max} - l_{\min} + 1)$ pairs of $(t, l)$ indices.
of interest. Adopt a standard order in column vector form as

\[
\begin{pmatrix}
(t_{\text{min}}, l_{\text{min}}) \\
(t_{\text{min}}, l_{\text{min}} + 1) \\
\vdots \\
(t_{\text{min}}, l_{\text{max}}) \\
(t_{\text{min}} + 1, l_{\text{min}}) \\
\vdots \\
(t_{\text{min}} + 1, l_{\text{max}}) \\
\vdots \\
(t_{\text{max}}, l_{\text{min}}) \\
\vdots \\
(t_{\text{max}}, l_{\text{max}})
\end{pmatrix}
\]

Then with rows and columns indexed by integers \( t' \) and \( t'' \) from \( t_{\text{min}} \) to \( t_{\text{max}} \), define the \((t_{\text{max}} - t_{\text{min}} + 1) \times (t_{\text{max}} - t_{\text{min}} + 1)\) matrix

\[
T = \begin{pmatrix}
\max(0, 1 + \alpha_2 |t' - t''|)
\end{pmatrix}
\]

Similarly, with rows and columns indexed by integers \( l' \) and \( l'' \) from \( l_{\text{min}} \) to \( l_{\text{max}} \) and letting \( \tilde{l} \) stand for the pseudo-wavelength corresponding to wavelength index \( l \), we define
the \((l_{\text{max}} - l_{\text{min}} + 1) \times (l_{\text{max}} - l_{\text{min}} + 1)\) matrix

\[
F = \begin{pmatrix}
\exp(-\beta_2(\text{reference})(\tilde{l} - \tilde{l}'^2))
\end{pmatrix}.
\] (36)

With this notation, the estimated correlation matrix for the \(\epsilon_2\) process over the \((t, l)\) pairs listed in display (34) is compactly written as \(T \otimes F\) (the Kronecker product of \(T\) and \(F\)).

Next, recall that the variances of the “white noise” quantities \(\epsilon_1\) are different at each \((t, l)\) pair and are estimated as \(\hat{\delta}^2(t, l)\). Then the estimated variance of the nuggetless component \(\epsilon_2\) must be \(1 - \hat{\delta}^2(t, l)\). For \((t, l)\) pairs in a region listed as in (34), let

\[
\Delta = \text{diag}(\hat{\delta}^2(t, l)).
\] (37)

Then the estimated covariance matrix for the nuggetless process \(\epsilon_2\) is

\[
(I - \Delta)^{\frac{1}{2}} T \otimes F(I - \Delta)^{\frac{1}{2}}
\] (38)

and for

\[
\epsilon(\tau, \lambda) = \epsilon_1(\tau, \lambda) + \epsilon_2(\tau, \lambda),
\] (39)

assuming the two components are independent, the estimated covariance matrix is

\[
(I - \Delta)^{\frac{1}{2}} T \otimes F(I - \Delta)^{\frac{1}{2}} + \Delta.
\] (40)

Then, with the estimated class variance function for the region of interest again \(\hat{\sigma}^2(t, l)\), let

\[
\Sigma = \text{diag}(\hat{\sigma}^2(t, l)).
\] (41)
The estimated covariance matrix for the values of $\tilde{S}(t + \hat{\eta}_k, l)$ in the given region is

$$\Sigma^2 \{(I - \Delta)^{1/2} T \otimes F(I - \Delta)^{1/2} + \Delta\} \Sigma^2 = D_1 [T \otimes F + D_2] D_1,$$

(42)

where diagonal matrices $D_1 = \Sigma^{1/2} (I - \Delta)^{1/2}$ and $D_2 = \text{diag} \left( \frac{\delta^2(t,l)}{1 - \delta^2(t,l)} \right)$ are specific to the $(t, l)$ pairs being considered.

So finally, we have completely specified a model for the burns of each of three types of safety matches, Class A, Class B, and Class C. A partial description of the particular models developed for these three classes of safety matches is given in the Appendix.

5.4 Modeling Summary

There were many steps involved in obtaining the final models. With some thought, these steps can be applied to model data that is collected from nearly any pseudo-imaging sensor operating in any part of the electromagnetic spectrum. Given below is a summary of the main steps we used to model our data once all preprocessing was complete.

The first step in estimating the model parameters was the time registration. The temporal shift parameter for each observation within each class was estimated according to (12). Proper alignment of all events used in a training set was key to determining the class mean and standard deviation functions. We note that time registration will also be important when models like this are used as the basis of discrimination and detection algorithms, as real events need not necessarily be observed from initiation.

After proper temporal alignment and all preprocessing was complete, the intensity
values at each time-wavelength pair were averaged acrossed all the observations within a class of data. The mean function represents the characteristic signature (for the preprocessed/transformed data) for the particular class of events and any observation from this class can be thought of as varying around the same characteristic signature.

An observation-specific “intensity shift” parameter $d_k$ was estimated as in (13). This allows for the modeling of objects from the same class of events observed at different distances. Recall that if two identical events were observed by the sensor through the same atmosphere at different distances, the event closer to the sensor would record higher intensity values.

Next, the variance surface was estimated using the characteristic signature (mean surface) and the time- and intensity-shifted (preprocessed/transformed) observations. This captured the many sources of variability including, but not limited to shot-to-shot, atmospheric interference, and sensor noise.

The last step in the modeling was determining appropriate forms for the correlation structure and estimating parameters for those functions. The process was complicated and only a sketch is provided here. (The details that were used in modeling correlations for classes of safety matches may need to be altered for other classes of point source events observed through similar sensors. However, the main ideas used here should be applicable.) The standarized residuals at each (time index-wavelength index) pair were used in the estimation of a correlation function in time and a separate correlation function in wavelength. Separate time and wavelength correlation functions were estimated for event initiation and steady-state phases and to the right and left
sides of the undeviated wavelength. The data from the four corresponding regions were treated as independent. Next the standardized error process was decomposed into two components, one “white noise” uncorrelated component and the other involving correlation between intensities at different \((t, l)\) pairs within the same observation. The variance function for the “white noise” was estimated for each \((t, l)\) region within each class of events. After an adjustment for the “white noise” was made to the standarized error at each \((\text{time index-wavelength index})\) pair, empirical variograms were calculated to determine the form of the correlation structure and the parameter estimates for those correlation structures.

By using this modeling methodology, information from every indexed time and wavelength has been retained and modeled. No data reduction was necessary or used before completely modeling the three classes of safety matches. We have effectively modeled both the characteristic signatures of the three event types and the observed variability from a variety of sources, including shot-to-shot, atmospheric interference, and sensor noise. Future work can use models like these in the development of detection and discrimination algorithms.

6 Simulating and Assessing Data from Fitted Models

We illustrate the effectiveness of our modeling by generating simulated sensor data from our fitted models and comparing it to the original (real) data. For a specified event class, data were simulated independently in the four regions indicated in Figure 8 and then put together to form a single data set.
Let $Q_1$ and $Q_2$ be independent, multivariate normal random vectors with $E(Q_1) = E(Q_2) = 0$ and

$$\text{Var}(Q_1) = T \otimes F$$  \hspace{1cm} (43)$$

and

$$\text{Var}(Q_2) = D_2$$  \hspace{1cm} (44)$$

where $T$, $F$, and $D_2$ are as defined in the previous section and are unique to a given region. (The class-specific estimated values for all parameters were used when simulating data.) $Q = Q_1 + Q_2$ was simulated for each sub-region independently. The final simulated data were generated as

$$D_1Q$$  \hspace{1cm} (45)$$

which is multivariate normal with mean 0 and covariance $D_1[T \otimes F + D_2]D_1$ as defined in equation (42). To obtain the data with correct mean, the values $\hat{\mu}(t, l)$ were added for each $(t, l)$ location. We denote a simulated data set by $\tilde{\theta}(t, l)$.

In simulating data, the software we used could not allocate enough memory to construct a covariance structure for the largest $(t, l)$ region. Therefore we simulated values only for every fourth time step to reduce the size of the matrix $T$ corresponding to each region. Figure 12 shows four realizations from Class A sampled at every fourth time index and Figure 13 shows corresponding simulated data. Visual comparison of the simulated data to the realizations suggests that our modeling has been effective. In the balance of this section we consider a more quantitative assessment of this effectiveness.

Recall that $\tilde{S}(t, l)$ represents preprocessed/transformed data, and that there are 20 realizations for Classes A and B and 17 realizations for Class C. Twenty datasets
were simulated from each of the three models. We compared realization-to-realization, simulation-to-simulation, and realization-to-simulation within a class as follows. After the realizations had been temporally and intensity shifted and simulations had been intensity shifted (there was no need for temporal alignment with the simulations), the absolute difference was found at each (time index-wavelength index) pair for each of the three kinds of comparisons. Let

\[ \Psi_{j,k}(t,l) = |\tilde{S}_j(t + \hat{\eta}_j, l) - \hat{d}_j - (\tilde{\theta}_k(t, l) - \hat{d}_k)| \text{ for all } j = 1, \ldots J, k = 1, \ldots K \]  

(46)

be the absolute value of the difference at time index \( t \) and wavelength index \( l \) between the \( j^{th} \) realization after temporal and intensity alignment and \( k^{th} \) simulated data after intensity alignment. Further, let

\[ \Phi_{j,k}(t,l) = |\tilde{S}_j(t + \hat{\eta}_j, l) - \hat{d}_j - (\tilde{S}_k(t + \hat{\eta}_k, l) - \hat{d}_k)| \text{ for all } j \neq k \]  

(47)

be the absolute value of the difference between the \( j^{th} \) and \( k^{th} \) realizations after temporal and intensity alignment. Last, let

\[ \Gamma_{j,k}(t,l) = |\tilde{\theta}_j(t, l) - \hat{d}_j - (\tilde{\theta}_k(t, l) - \hat{d}_k)| \text{ for all } j \neq k \]  

(48)

be the absolute value of the difference between the \( j^{th} \) and the \( k^{th} \) simulations after intensity alignment.

To make meaningful comparisons between the realized and simulated datasets, \( \Psi \), \( \Phi \), and \( \Gamma \) had to be reduced to something that was easily handled and would lend itself easily to interpretation. Further, for comparisons between classes, this reduction needed to not be sensitive to the number of time steps indices for a given class.
So let $\Psi_{j,k}^p$, $\Phi_{j,k}^p$, and $\Gamma_{j,k}^p$ represent the $p^{th}$ percentile across all time index-wavelength index pairs of $\Psi_{j,k}$, $\Phi_{j,k}$, and $\Gamma_{j,k}$, respectively. Then consider the sets of all these $p^{th}$ percentiles

$$\psi^p = \{\Psi_{j,k}^p | j = 1, \ldots, J, k = 1, \ldots, K\}$$

$$\phi^p = \{\Phi_{j,j'}^p | j = 1, \ldots, J - 1, j' = 2, \ldots, J, j < j'\}$$

and

$$\gamma^p = \{\Gamma_{k,k'}^p | k = 1, \ldots, K - 1, k' = 2, \ldots, K, k < k'\}.$$

The empirical distributions of $\psi^p$, $\phi^p$, and $\gamma^p$ for each class were observed together for various choices of $p$. The top row of plots in Figures (14), (15), and (16) show these empirical distribution functions plotted together for the 5th, 50th, and 95th percentiles, respectively. These can be thought of as portraying respectively how far apart a pair of waterfall plots/data sets tend to be in the regions where they are most alike, differ by more or less an average amount, and in the regions where they are most different.

In all these plots, the empirical distribution of $\gamma$ is slightly shifted to the left of $\phi$ and this is most apparent in the plot of the 5th percentiles as the range of values on the horizontal axis is small, amplifying the difference. These empirical distributions ($\gamma$) are only slightly shifted to the left. This pattern was consistent across all the percentiles examined, and suggests that the simulated datasets are perhaps slightly smoother than the actual realizations. This is not surprising, as one would not expect a fitted model to capture every detail of the realizations. However, since the shift is not too extreme for any percentile, we feel the modeling has done an adequate job of describing the realizations.
Now, as mentioned earlier, the Class A and Class B matches are very similar in chemical composition, as Class B matches are simply two Class A matches physically connected. Class C matches have a very different chemical composition than the other two classes of matches. We also investigated how the realizations and final models compared between classes. Let $\tilde{S}_{A}(t, l)$ be a realization for Class A and $\tilde{S}_{B}(t, l)$ be a realization from Class B. Further, let $\tilde{\theta}_{A}(t, l)$ be a simulated dataset from the model for Class A and $\tilde{\theta}_{B}(t, l)$ be a simulated dataset from the model for Class B. Now, comparisons can be made between realizations from two different classes or between simulated datasets from the two different classes, or between a realization from one class and a simulated dataset from a second class. More completely, let

$$K_{j,k}(t, l) = |\tilde{S}_{A,j}(t + \hat{\eta}_{A,j}, l) - d_{A,j} - (\tilde{S}_{B,k}(t + \hat{\eta}_{B,k}, l) - d_{B,k})|$$  \hspace{1cm} (49)$$

for all $j = 1, \ldots, J$ and $k = 1, \ldots, K$ be the absolute value of the difference at time index $t$ and wavelength index $l$ between the $j^{th}$ realization from Class A and $k^{th}$ realization from Class B after temporal and intensity alignment. Let

$$X_{j,k}(t, l) = |\tilde{\theta}_{A,j}(t + \hat{\eta}_{A,j}, l) - \hat{d}_{A,j} - (\tilde{\theta}_{B,k}(t + \hat{\eta}_{B,k}, l) - \hat{d}_{B,k})|$$  \hspace{1cm} (50)$$

for all $j = 1, \ldots, J$ and $k = 1, \ldots, K$ be the absolute value of the difference at time index $t$ and wavelength index $l$ between the $j^{th}$ simulated dataset from Class A and $k^{th}$ simulated dataset from Class B after temporal and intensity alignment. Note the temporal alignment is needed here for the simulated data as the “peak” in the mean function is not at the same location for each class. Further, let

$$\Omega_{j,k}(t, l) = |\tilde{S}_{A,j}(t + \hat{\eta}_{A,j}, l) - \hat{d}_{A,j} - (\tilde{\theta}_{B,k}(t + \hat{\eta}_{B,k}, l) - \hat{d}_{B,k})|$$  \hspace{1cm} (51)$$
for all \( j = 1, \ldots, J \) and \( k = 1, \ldots, K \) be the absolute value of the difference at time index \( t \) and wavelength index \( l \) between the \( j^{th} \) realizations from Class A and the \( k^{th} \) simulated dataset from Class B after temporal and intensity alignment. Last, let

\[
Z_{j,k}(t,l) = |\tilde{\theta}_{A,j}(t + \hat{\eta}_{A,j}, l) - \hat{d}_{A,j} - (\tilde{S}_{B,k}(t + \hat{\eta}_{B,k}, l) - \hat{d}_{B,k})| \tag{52}
\]

for all \( j = 1, \ldots, J \) and \( k = 1, \ldots, K \) be the absolute value of the difference at time index \( t \) and wavelength index \( l \) between the \( j^{th} \) simulated dataset from Class A and the \( k^{th} \) realization from Class B after temporal and intensity alignment. As before, the \( p^{th} \) percentile of these differences can be examined. Let

\[
k^p = \{K^p_{j,k} | j = 1, \ldots, J, k = 1, \ldots, K\},
\]

\[
\chi^p = \{X^p_{j,k} | j = 1, \ldots, J, k = 1, \ldots, K\},
\]

\[
\omega^p = \{\Omega^p_{j,k} | j = 1, \ldots, J, k = 1, \ldots, K\},
\]

and

\[
\zeta^p = \{Z^p_{j,k} | j = 1, \ldots, J, k = 1, \ldots, K\}.
\]

Similar between-class comparisons can be computed for Class A and Class C, and Class B and Class C.

The between-class comparisons are shown on the bottom row of plots in Figures (14), (15), and (16). (Again, distributions of 5\(^{th}\), 50\(^{th}\), and 95\(^{th}\) percentiles represent how far apart pairs of datasets are where they are most alike, differ by more or less an average amount, and where they are least alike.) The between-class comparisons for Class A and Class B take values very similar to those for the within-class comparisons for either Class A or Class B. This is what one would expect since these classes are very similar to one another in chemical composition. Further, the distribution functions for the between-class
comparisons between Class C and either Class A or Class B place more weight on larger values. This is also to be expected since Class C is potentially very different in chemical composition from the other two classes. Again, the simulation-to-simulation comparisons take values slightly smaller than the realization-to-realization differences even if the comparison is made between classes, likely indicating slightly smoother simulations than realized datasets. The comparisons of one realization to one simulated dataset from two different classes are also slightly smaller in value that the realization-to-realization comparisons. Further, perhaps the most disparity between these distribution functions exist when comparisons are made with Class C.

The data models we have developed here are both subtle and complex. While they may not capture all real variability of the processes, Figures 12 through 16 show that they can be used to generate data that reasonably represent characteristics common to each class, and within- and among-class variability.
Figure 12  Four Real Observed Datasets From Class A
Figure 13  Four Simulated Datasets for Class A
Comparing Classes Using 5th Percentile of Absolute Difference

Figure 14 Within- and Between-Class Comparisons Using 5\textsuperscript{th} Percentile of Absolute Differences
Comparing Classes Using Median of Absolute Difference

Figure 15  Within- and Between-Class Comparisons Using 50th Percentile of Absolute Differences
Comparing Classes Using 95th Percentile of Absolute Difference

Figure 16  Within- and Between-Class Comparisons Using 95th Percentile of Absolute Differences
7 Conclusions

We have modeled data from three classes of energetic electromagnetic events. A major goal was to model spectral-temporal data of this nature without first reducing the dimensionality of the data, i.e. to develop a model for intensities at all available \((t, l)\) pairs.

The data has been modeled using relatively few parameters. With the use of product correlations, the number of parameters needed to model the potentially large number of sources of variability has been minimized. A temporal alignment parameter has also been included in the model. With this model element, algorithms can be developed that have the potential to cleanly handle the time registration problem if an incoming data stream is not observed from event initiation.

The models developed in this paper are intended for use as the basis for a decision algorithm that can be used to distinguish between different energetic events in real time. Three models can be used to assess the likelihood that an incoming data stream matches one of the previously modeled electromagnetic energetic event classes.

8 Acknowledgments

We thank the Electromagnetic Technology Branch of the Sensors Directorate of Air Force Research Lab (AFRL/SNHI) for funding this project. The funding was administered by Solid State Scientific Corporation.
Appendix

The following are the details for equation (19).

\[
\begin{align*}
\text{Var}(Y_k(t, l)) &= \text{Var}(\hat{\xi}_k(t, l)) \\
&\quad - \frac{1}{2} \{ \text{Var}(\hat{\xi}_k(t, l - 1)) + \text{Var}(\hat{\xi}_k(t, l + 1)) \\
&\quad + \text{Var}(\hat{\xi}_k(t - 1, l)) + \text{Var}(\hat{\xi}_k(t + 1, l)) \} \\
&\quad + \frac{1}{8} \{ \text{Cov}(\hat{\xi}_k(t, l), \hat{\xi}(t, l - 1)) + \text{Cov}(\hat{\xi}_k(t, l), \hat{\xi}_k(t, l + 1)) \\
&\quad + \text{Cov}(\hat{\xi}_k(t, l), \hat{\xi}_k(t - 1, l)) + \text{Cov}(\hat{\xi}_k(t, l), \hat{\xi}_k(t + 1, l)) \} \\
&\quad + \frac{1}{8} \{ \text{Cov}(\hat{\xi}_k(t, l - 1), \hat{\xi}_k(t, l + 1)) + \text{Cov}(\hat{\xi}_k(t, l - 1), \hat{\xi}_k(t - 1, l)) \\
&\quad + \text{Cov}(\hat{\xi}_k(t, l - 1), \hat{\xi}_k(t + 1, l)) \} \\
&\quad + \frac{1}{8} \{ \text{Cov}(\hat{\xi}_k(t, l + 1), \hat{\xi}_k(t - 1, l)) + \text{Cov}(\hat{\xi}_k(t, l + 1), \hat{\xi}_k(t + 1, l)) \} \\
&\quad + \frac{1}{8} \{ \text{Cov}(\hat{\xi}_k(t - 1, l), \hat{\xi}_k(t + 1, l)) \}
\end{align*}
\]
\[ \approx \text{Var}(\epsilon_k(\tau, \lambda)) \]
\[- \frac{1}{2} \left\{ \text{Var}(\epsilon_k(\tau, \lambda - 1)) + \text{Var}(\epsilon_k(\tau, \lambda + 1)) \\
+ \text{Var}(\epsilon_k(\tau - 1, \lambda)) + \text{Var}(\epsilon_k(\tau + 1, \lambda)) \right\} \]
\[ + \frac{1}{8} \left\{ \text{Cov}(\epsilon_k(\tau, \lambda), \epsilon_k(\tau, \lambda - 1)) + \text{Cov}(\epsilon_k(\tau, \lambda), \epsilon_k(\tau, \lambda + 1)) \\
+ \text{Cov}(\epsilon_k(\tau, \lambda), \epsilon_k(\tau - 1, \lambda)) + \text{Cov}(\epsilon_k(\tau, \lambda), \epsilon_k(\tau + 1, \lambda)) \right\} \]
\[ + \frac{1}{8} \left\{ \text{Cov}(\epsilon_k(\tau, \lambda - 1), \epsilon_k(\tau, \lambda)) + \text{Cov}(\epsilon_k(\tau, \lambda - 1), \epsilon_k(\tau - 1, \lambda)) \\
+ \text{Cov}(\epsilon_k(\tau, \lambda), \epsilon_k(\tau + 1, \lambda)) \right\} \]
\[ + \frac{1}{8} \left\{ \text{Cov}(\epsilon_k(\tau, \lambda + 1), \epsilon_k(\tau - 1, \lambda)) + \text{Cov}(\epsilon_k(\tau, \lambda + 1), \epsilon_k(\tau + 1, \lambda)) \right\} \]
\[ + \frac{1}{8} \left\{ \text{Cov}(\epsilon_k(\tau - 1, \lambda), \epsilon_k(\tau + 1, \lambda)) \right\} \]
\[ = 1 + \frac{1}{16} \tag{4} \]
\[- \frac{1}{2} \sqrt{1 - \delta_k^2(\tau, \lambda)} \left\{ R_\tau(0) R_\lambda(1) \sqrt{1 - \delta_k^2(\tau, \lambda - 1)} \\
+ R_\tau(0) R_\lambda(1) \sqrt{1 - \delta_k^2(\tau, \lambda + 1)} \\
+ R_\tau(1) R_\lambda(1) \sqrt{1 - \delta_k^2(\tau - 1, \lambda)} \\
+ R_\tau(1) R_\lambda(0) \sqrt{1 - \delta_k^2(\tau + 1, \lambda)} \right\} \]
\[ + \frac{1}{8} \sqrt{1 - \delta_k^2(\tau, \lambda - 1)} \left\{ R_\tau(0) R_\lambda(2) \sqrt{1 - \delta_k^2(\tau, \lambda + 1)} \\
+ R_\tau(1) R_\lambda(0) \sqrt{1 - \delta_k^2(\tau - 1, \lambda)} \\
+ R_\tau(1) R_\lambda(1) \sqrt{1 - \delta_k^2(\tau + 1, \lambda)} \right\} \]
\[ + \frac{1}{8} \sqrt{1 - \delta_k^2(\tau, \lambda + 1)} \left\{ R_\tau(1) R_\lambda(1) \sqrt{1 - \delta_k^2(\tau - 1, \lambda)} \\
+ R_\tau(1) R_\lambda(1) \sqrt{1 - \delta_k^2(\tau + 1, \lambda)} \right\} \]
\[ + \frac{1}{8} \sqrt{1 - \delta_k^2(\tau - 1, \lambda)} \left\{ R_\tau(2) R_\lambda(0) \sqrt{1 - \delta_k^2(\tau + 1, \lambda)} \right\} \]

By definition, \( R_\tau(0) = 1 \) and \( R_\lambda(0) = 1 \) and if one assumes all of \( R_\tau(1), \ R_\tau(2), \ R_\lambda(1), \ R_\lambda(2) \) are approximately 1, and \( \delta^2(\tau, \lambda) \approx \delta^2_k(\tau - 1, \lambda) \approx \delta^2_k(\tau, \lambda - 1) \approx \delta^2_k(\tau, \lambda + 1) \), the above
equation reduces to

\[
\operatorname{Var}(Y(t,l)) \approx \frac{5}{4} - 2(1 - \delta^2_k(\tau, \lambda)) + \frac{3}{4}(1 - \delta^2_k(\tau, \lambda))
\]

\[
= \frac{5}{4} \delta^2_k(\tau, \lambda).
\]
Figure A.1: Class A Mean and Standard Deviation Surfaces

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Left Early</th>
<th>Right Early</th>
<th>Left Late</th>
<th>Right Late</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>$-2.2 \times 10^{-2}$</td>
<td>$-1.05 \times 10^{-2}$</td>
<td>$-5.0 \times 10^{-4}$</td>
<td>$-2.0 \times 10^{-3}$</td>
</tr>
<tr>
<td>$\beta_{2(\text{reference})}$</td>
<td>$1.8 \times 10^{-3}$</td>
<td>$1.10 \times 10^{-3}$</td>
<td>$1.2 \times 10^{-4}$</td>
<td>$2.4 \times 10^{-3}$</td>
</tr>
<tr>
<td>$\tilde{t}_{\text{low}}$</td>
<td>$2.345 \times 10^{-1}$</td>
<td></td>
<td></td>
<td>$3.162$</td>
</tr>
<tr>
<td>$\tilde{t}_{\text{middle}}$</td>
<td>$0.802$</td>
<td>$0.606$</td>
<td>$0.274$</td>
<td>$0.603$</td>
</tr>
<tr>
<td>$\tilde{t}_{\text{down}}$</td>
<td>$5.262$</td>
<td>$1.658$</td>
<td>$2.928$</td>
<td>$1.732$</td>
</tr>
</tbody>
</table>

Table A.1: Class A Model Parameter Estimates
Figure A.2: Class B Mean and Standard Deviation Surfaces

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Left Early</th>
<th>Right Early</th>
<th>Left Late</th>
<th>Right Late</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>$-1.2 \times 10^{-1}$</td>
<td>$-1.18 \times 10^{-1}$</td>
<td>$-6.0 \times 10^{-4}$</td>
<td>$-8.0 \times 10^{-4}$</td>
</tr>
<tr>
<td>$\beta_{2(\text{reference})}$</td>
<td>$9.0 \times 10^{-4}$</td>
<td>$1.0 \times 10^{-4}$</td>
<td>$3.3 \times 10^{-5}$</td>
<td>$1.6 \times 10^{-3}$</td>
</tr>
<tr>
<td>$l_{\text{low}}$</td>
<td></td>
<td>0.707</td>
<td></td>
<td>3.162</td>
</tr>
<tr>
<td>$l_{\text{middle}}$</td>
<td>0.452</td>
<td>0.151</td>
<td>0.173</td>
<td>0.444</td>
</tr>
<tr>
<td>$l_{\text{down}}$</td>
<td>5.000</td>
<td>0.500</td>
<td>1.483</td>
<td>1.333</td>
</tr>
</tbody>
</table>

Table A.2: Class B Model Parameter Estimates
Figure A.3: Class C Mean and Standard Deviation Surfaces

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Left Early</th>
<th>Right Early</th>
<th>Left Late</th>
<th>Right Late</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>$-9.0 \times 10^{-4}$</td>
<td>$-7.8 \times 10^{-3}$</td>
<td>$-6.0 \times 10^{-4}$</td>
<td>$-8.0 \times 10^{-4}$</td>
</tr>
<tr>
<td>$\beta_{2(\text{reference})}$</td>
<td>$6.0 \times 10^{-5}$</td>
<td>$1.1 \times 10^{-3}$</td>
<td>$1.17 \times 10^{-4}$</td>
<td>$3.9 \times 10^{-3}$</td>
</tr>
<tr>
<td>$l_{\text{low}}$</td>
<td>0.782</td>
<td></td>
<td></td>
<td>1.225</td>
</tr>
<tr>
<td>$l_{\text{middle}}$</td>
<td>0.866</td>
<td>0.920</td>
<td>1.217</td>
<td>1.363</td>
</tr>
<tr>
<td>$l_{\text{down}}$</td>
<td>0.816</td>
<td>1.106</td>
<td>0.917</td>
<td>2.550</td>
</tr>
</tbody>
</table>

Table A.3: Class C Model Parameter Estimates
Figure A.4: Class A $\delta^2$ Surfaces
Figure A.5: Class B $\delta^2$ Surfaces
Figure A.6: Class C $\delta^2$ Surfaces
9 References


CHAPTER 2. AN ALGORITHM FOR REAL-TIME DISCRIMINATION BASED ON SPECTRAL-TEMPORAL DATA FROM POINT SOURCE EVENTS

Monica Reising, Max Morris, Stephen Vardeman, and Shawn Higbee

1 Abstract

This article discusses the preliminary design of an algorithm for real-time discrimination between different types of point source events based on spectral-temporal data. The properties, including rate of false rejections and average times to rejection, were assessed using both simulated data from fitted models for 3 classes of safety matches and observed data from the same classes of safety matches. This work shows promise as an initial step towards implementing such an algorithm for real-time discrimination, but there are some practical concerns that must be addressed in future work.

2 Introduction

Reising et al.(2008) showed that data collected using a pseudo-imager with ignition and burn data for 3 classes of safety matches were appropriately described using Gaussian random field models. That paper discussed the statistical modeling of spectral-temporal data from point source events collected under controlled experimental conditions. Here we use the models in Reising et al.(2008) as a test bed for the development of an
algorithm for determining in real time whether an incoming data stream is consistent with the model of any one of the three classes of safety matches.

The collection of information (mean, variance, and correlation) stored for each class of safety matches will henceforth be referred to as a *library*. This paper will briefly discuss the modeling of energy from point source events, and then proceed to the development of a discrimination algorithm, selection of cut-points for determining if an incoming data stream is inconsistent with a class of events stored in the library, and operating characteristics of the method (including false rejection rates and, given rejection, the average time to rejection) based on a simulation study. The results of applying the algorithm to the data used to create the models stored in the library as well as data obtained from other point source events will be discussed.

3 Modeling Background

Reising *et al.* (2008) described an extensive analysis for intensity data indexed both by time, $\tau$, and wavelength, $\lambda$. Briefly, they showed that a plausible model for the $k^{th}$ observation, $\tilde{\theta}_k(\tau, \lambda)$, of transformed burn data from an experiment using safety matches (the test bed for model development) had a decomposition as

$$\tilde{\theta}_k(\tau, \lambda) = d_k + \mu(\tau + \eta_k \Delta \tau, \lambda) + \sigma(\tau + \eta_k \Delta \tau, \lambda)\epsilon(\tau, \lambda) \quad \forall \tau \in (0, T) \text{ and } \lambda \in \Lambda \quad (1)$$

where $\Lambda$ represents the set of wavelengths observed by the sensor and $\Delta \tau$ is the time between two consecutive sampling points. $d_k$ and $\eta_k$ are intensity and time shift parameters, respectively, associated with the $k^{th}$ event of a particular class of events. $\mu$ and $\sigma$ are the class mean and standard deviation functions and $\epsilon$ is a “spatial” stochastic
process with mean zero, variance 1, and has a correlation structure specific to the class of events. Models of form (1) were developed for three different classes of safety matches.

The data analyzed by Reising et al. (2008) were collected using a sensor that produces intensity values on a discrete grid of times and wavelengths, indexed here for convenience by corresponding integers \( t = 0, 1, \ldots \) and \( l = 1, 2, \ldots, 149 \) and denoted \( S(t, l) \). Following their notation, let \( \tilde{S}(t, l) \) denote spectral-temporal intensity data that have been transformed according to

\[
\tilde{S} = \begin{cases} 
\ln(S) & \text{for } \theta \geq 2 \\
S(\ln(2)/2) & \text{for } -2 < \theta < 2 \\
-\ln(|S|) & \text{for } \theta \leq -2.
\end{cases}
\]  

(2)

Multiple realizations of transformed data, \( \tilde{S}(t, l) \) were used to estimate the parameters of model (1) for each class of safety matches. A fitted mean function and an estimated covariance function were stored in the library for each class of safety matches.

While model (1) in theory allows for time and wavelength to be continuous, the mean function stored in the library is discrete in time and wavelength corresponding to the spectral and temporal sampling rates of the sensor. Further, the algorithm development described in this article was based on a subset of the available time and wavelength values. The mean function stored in the library corresponding to the appropriate subset (in time and wavelength) will be denoted as \( \mu(t, l) \) and the spectral-temporal covariance matrix as \( \Xi \). It is worth noting that the number of time steps modeled for each class is different and therefore, the dimensionality of \( \Xi \) varies class-to-class (Reising et al. 2008).
4 Development of a Discrimination Algorithm

The algorithm described in this paper is designed to determine in real time if an incoming data stream is consistent with one of the previously modeled classes of safety matches. Within the limits of our experimental data, the hope is that this algorithm will allow for the differentiation between the 3 types of safety matches. The goal is to be able to quickly rule out the classes to which an incoming data stream cannot belong, while maintaining a small rate of false rejection for the class to which the incoming data belong.

In real applications it is not necessarily the case that a point source event will always be observed from time of event initiation. Therefore, a real-time discrimination algorithm needs to be designed to be effective even when the earliest data observed correspond to mid-event dynamics. To this end, let the transformed data available to the algorithm at any given time, corresponding to \( m + 1 \) consecutive sampling times be denoted \( \tilde{S}(t, l) \), with \( t = t_{\text{lower}}, ldots, t_{\text{lower}} + m \) and \( L = 1, \ldots, L \), where \( L \) is the number of wavelength indices available. Then for a specified class mean function \((\mu)\), time index \((t')\), and
intensity shift value \((d)\), define:

\[
y_{t',d} = \begin{pmatrix}
\tilde{S}(t_{lower}, 1) - d - \mu(t', 1) \\
\tilde{S}(t_{lower}, 2) - d - \mu(t', 2) \\
\vdots \\
\tilde{S}(t_{lower}, L) - d - \mu(t', L) \\
\tilde{S}(t_{lower} + 1, 1) - d - \mu(t' + 1, 1) \\
\vdots \\
\tilde{S}(t_{lower} + 1, L) - d - \mu(t' + 1, L) \\
\vdots \\
\tilde{S}(t_{lower} + m, 1) - d - \mu(t' + m, 1) \\
\vdots \\
\tilde{S}(t_{lower} + m, L) - d - \mu(t' + m, L)
\end{pmatrix}.
\]

Let \(N_t\) be the total number of time steps modeled for a particular class, so \(m = 0, 1, \ldots, N_t - t_{lower} + 1\) and \(t' = 0, 1, \ldots, N_t\). Note that in practice, we do not know which time point in the library (indexed by \(t'\)) actually physically corresponds to the first time point in the data stream (indexed by \(t_{lower}\)). Further, let \(\Xi(t', m)\) represent the \((m + 1)L \times (m + 1)L\) portion of the covariance matrix corresponding to the appropriate time frame(s). The statistic that will be used in the discrimination problem for the \(j^{th}\) event class when \((m + 1)\) consecutive time-frames of data are in hand will be

\[
D_j = \min_{t',d} \ y_{t',d}^{'} \Xi(t', m)^{-1} \ y_{t',d}.
\]

Minimization is over allowable integer values of \(t'\), to affect time registration between the signal and library, and real-valued \(d\), to allow for an overall correction for signal
strength.

Large values of this statistic (relative to what realizations from the model for class $j$ would produce) will constitute evidence that the incoming data stream does not belong to class $j$.

4.1 Determining Cut-Points for a Discrimination Algorithm

In determining cut-points (unique to each class in the library) used as a basis for making a real-time decision concerning whether an incoming data stream can be excluded as not generated by the model for class $j$, 1000 data sets were simulated from the models corresponding to each class of safety matches. Let $S_j^*(t, l)$ represent a simulated data set from the $j^{th}$ class,

$$S_j^* \sim \text{MVN} (\mu_j(t, l), \Xi_j)$$

(5)

where $S_j^*(t, l)$ is indexed by some trimmed set of times and wavelengths. (In the spirit of this paper providing a proof of concept and because the statistical software used in our study could not allocate enough memory to simulate the full data sets, a reduced dimensionality was used.) We sampled every 15th time step in both directions from the time index giving the largest total (across wavelength) intensity for a given class. This was done to try to assure that fair comparisons were being made. Additionally, the pixels in the focal plane array were also sampled in regular fashion so 8 pixels were chosen on each side (corresponding to shorter and longer wavelengths) of the so-called “undeivated wavelength” ($l = 80$). Therefore, in the balance of the discussion here, we consider $t = 1, 2, \ldots, N_t$ where $N_t$ corresponds to $\frac{1}{15}$ the original number of time steps
in the final models of Reising et al. (2008) and \( l = 1, 2, \ldots, 16 \) will correspond to the 16 (regularly spaced) pixels from the focal plane array representing a “thinned” spectrum of wavelengths.

Notice that the behavior of \( D_j \) should change as the number of observed frames of data increases. Therefore, a different cut-point would be appropriate for each value of \( m \) within each class. The first step in determining an appropriate cut-point was investigating the behavior of \( D_j \) for different numbers of consecutive frames \((m + 1)\) and the frame in which observation began \((t_{lower})\). Recall that an incoming data stream may not correspond with event initiation. Cut-points used in the discrimination need to result in decisions with small false rejection rates, but when appropriate not only reject membership in a class from the library, but also do it quickly. In that spirit, a determination of the appropriate cut-points was made.

Let \( D_{j,k}(t, n_t) \) represent the observed statistic from the \( k^{th} \) simulated data set, \( S_{j,k}^* \), from the \( j^{th} \) event class observed starting at time index \( t \) for a total of \( n_t \) time frames. The collections of \( D_{j,k}(t, n_t) \) is for values of \( n_t = 1, \ldots, N_t \),

\[
D_{j,k}(n_t) = \begin{pmatrix}
D_{j,k}(t_{lower}, n_t) \\
D_{j,k}(t_{lower} + 1, n_t) \\
\vdots \\
D_{j,k}(t_{lower} + m, n_t)
\end{pmatrix}
\]

(6)

were used in the determination of the appropriate cut-points.

Note the size of \( D_{j,k}(n_t) \) decreases as \( n_t \) increases because the value of \( m \) decreases. That is, as the number of observed time frames increases, there are fewer possible
starting places for the observed data frames. Regardless of which index \( t_{lower} \) corresponds to, if there is only one available time frame, it could correspond to any value of \( t = 1, 2, \ldots, N_t \). However, if there are \( n_t \) frames of data available for comparison to the library, the first observed data must correspond to \( t = 1 \) (and it is known that \( t_{lower} = 1 \)).

Let \( MD_{j,k}(n_t) = \max D_{j,k}(n_t) \) be the maximum value in \( D_{j,k}(n_t) \) computed from the \( k^{th} \) simulated realization, and denote the entire collection of these values (over all 1000 simulations from class \( j \)) by

\[
MD_j(n_t) = \begin{pmatrix}
MD_{j,1}(n_t) \\
MD_{j,2}(n_t) \\
\vdots \\
MD_{j,1000}(n_t)
\end{pmatrix}
\]

That is, \( MD_j(n_t) \) is the collection of maximum observed minimum values for the quadratic form in (4) for each value of \( n_t \). Cut-points (for judging the sizes of observed values of (4) based on a data record comprised of observations from \( n_t \) time periods) were chosen as the respective 99.5 percentiles of the sets \( MD_j(n_t) \). Let \( CP_j \) represent the vector of cut-points for the \( j^{th} \) class from the library. The length of \( CP_j \) depends on the number of time steps modeled and stored in the library. Hence an observed value \( D_j \) based on data from 5 consecutive time steps of data will be declared incompatible with the modeled characterization of class \( j \) if it is larger than the 5\(^{th} \) element of \( CP_j \).
4.2 Assessing the Performance of the Discrimination Algorithm

Once cut-points to use in the discrimination algorithm were determined for each class of safety matches, attention turned to assessing performance characteristics of the algorithm when applied to simulated data from the model for each class. An additional 1000 data sets were simulated according to (5) for each of the 3 classes of safety matches represented in the library. First, it was important to have small rejection rates for comparisons to the class from which a record was simulated. Tables 1-3 show the numbers of simulated data sets that were rejected as not appearing to be consistent with the characteristics of the class as represented in the library for both the 1st set of 1000 simulations (used in determining the cut-points) and the 2nd set of 1000 newly simulated data sets for each class. These counts refer to rejections made somewhere in the series of comparisons of first one frame of data to the stored class characteristics via (4), then two frames of data, etc., where only the part of the data beginning at the time step given in the left column is considered. In each case, no more than 9 false rejections occurred in the first set of 1000 simulations and no more than 13 occurred in the second.

In addition to having a small false rejection rate for each class the algorithm should also have a high rejection rate and short average time to rejection for observations from other classes. Tables 4-6 show the numbers of data sets (of the 2000 total simulated for the other classes) that were (at least eventually) classified ‘not matching’ a given set of class characteristics. Here, rejection occurs at a much higher rate (as it should) as high as 100 percent for data streams that begin close to event initiation. The tables also provide the average time to rejection based on those cases that led to rejection.
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected in 1000 Simulated Data Sets Used to Set Cut-Points</th>
<th>Number Rejected in 1000 Additional Data Sets</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>18</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>23</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>26</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>28</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>31</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>32</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>33</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>34</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1  Class A False Rejections
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected in 1000 Simulated Data Sets Used to Set Cut-Points</th>
<th>Number Rejected in 1000 Additional Data Sets</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>18</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>19</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>21</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>22</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>23</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>26</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>28</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>30</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>31</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>32</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>33</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>35</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2  Class B False Rejections
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected in 1000 Simulated Data Sets Used to Set Cut-Points</th>
<th>Number Rejected in 1000 Additional Data Sets</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>14</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>18</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3  Class C False Rejections
4.3 Assessing the Performance of the Discrimination Algorithm on Real Data

We also considered the performance of the discrimination algorithm on the transformed observed data from which our models were fitted. There were 20 observations from each of Class A and Class B that were used in the estimation of the model parameters. There were 17 observations from Class C that was used in the estimation of Class C model parameters. Table 7 shows the number of observed (preprocessed-transformed) data sets that were falsely rejected for each possible starting point in the signature. The false rejection rates are much higher than would allow for real application of this discrimination algorithm in its current form. However, Tables 1 - 3 display much smaller false rejection rates. The obvious implication is that the real data contain some features not fully captured by the modeling of Reising et al. (2008). That suggests that before actual deployment of the discrimination algorithm either more delicate modeling or alterations to make the algorithm more robust are needed. However, when increasing the cut-points by a factor of 4 the false reject rates drastically decrease. The number of false rejects range between 0 and 6 under these conditions. This suggests that while modeling of the event classes is not perfect, it is also not totally inadequate.
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected From Class B</th>
<th>Average Run Length To Reject Class B</th>
<th>Number Rejected From Class C</th>
<th>Average Run Length To Reject Class C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2000</td>
<td>1.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2000</td>
<td>1.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2000</td>
<td>1.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>2.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1999</td>
<td>2.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>1997</td>
<td>2.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1988</td>
<td>4.1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1967</td>
<td>5.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>1905</td>
<td>6.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>1807</td>
<td>7.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>1405</td>
<td>12.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>1260</td>
<td>12.7</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>1140</td>
<td>12.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>971</td>
<td>13.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>812</td>
<td>12.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>744</td>
<td>12.5</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>690</td>
<td>12.3</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>608</td>
<td>12.1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>508</td>
<td>11.7</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>436</td>
<td>11.5</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>412</td>
<td>10.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>381</td>
<td>10.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>267</td>
<td>9.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>180</td>
<td>9.3</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>25</td>
<td>121</td>
<td>8.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>26</td>
<td>87</td>
<td>7.5</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>79</td>
<td>6.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>68</td>
<td>5.7</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>52</td>
<td>4.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>30</td>
<td>50</td>
<td>3.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>31</td>
<td>29</td>
<td>3.1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>32</td>
<td>21</td>
<td>2.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>33</td>
<td>17</td>
<td>1.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>1.0</td>
<td>2000</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4  Results for Simulated Data from Class A Compared to Class B and Class C Models
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected From Class A</th>
<th>Average Run Length To Reject Class A</th>
<th>Number Rejected From Class C</th>
<th>Average Run Length To Reject Class C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2000</td>
<td>3.3</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2000</td>
<td>6.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2000</td>
<td>6.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2000</td>
<td>6.7</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>2000</td>
<td>6.3</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>2000</td>
<td>5.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>2000</td>
<td>4.7</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>2000</td>
<td>5.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>2000</td>
<td>5.1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>2000</td>
<td>5.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>2000</td>
<td>5.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>2000</td>
<td>5.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>2000</td>
<td>5.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>2000</td>
<td>6.3</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>1999</td>
<td>7.1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>1997</td>
<td>7.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>1995</td>
<td>8.2</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>1995</td>
<td>8.2</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>1990</td>
<td>8.0</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>1986</td>
<td>7.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>1983</td>
<td>7.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>1972</td>
<td>7.3</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>1974</td>
<td>6.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>1959</td>
<td>6.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>25</td>
<td>1956</td>
<td>6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>26</td>
<td>1934</td>
<td>5.4</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>27</td>
<td>1917</td>
<td>4.9</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>28</td>
<td>1907</td>
<td>4.5</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>29</td>
<td>1890</td>
<td>4.2</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>30</td>
<td>1827</td>
<td>3.8</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>31</td>
<td>1695</td>
<td>3.2</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>32</td>
<td>1527</td>
<td>2.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>33</td>
<td>1335</td>
<td>2.1</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>34</td>
<td>1018</td>
<td>1.6</td>
<td>2000</td>
<td>1</td>
</tr>
<tr>
<td>35</td>
<td>597</td>
<td>1.0</td>
<td>2000</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 5: Results for Simulated Data from Class B Compared to Class A and Class C Models
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected From Class A</th>
<th>Average Run Length To Reject Class A</th>
<th>Number Rejected From Class B</th>
<th>Average Run Length To Reject Class B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2000</td>
<td>7.7</td>
<td>2000</td>
<td>5.2</td>
</tr>
<tr>
<td>2</td>
<td>1998</td>
<td>8.7</td>
<td>2000</td>
<td>6.2</td>
</tr>
<tr>
<td>3</td>
<td>1984</td>
<td>8.8</td>
<td>1998</td>
<td>6.5</td>
</tr>
<tr>
<td>4</td>
<td>1933</td>
<td>8.7</td>
<td>1990</td>
<td>6.5</td>
</tr>
<tr>
<td>5</td>
<td>1824</td>
<td>8.7</td>
<td>1960</td>
<td>6.4</td>
</tr>
<tr>
<td>6</td>
<td>1586</td>
<td>8.1</td>
<td>1912</td>
<td>6.1</td>
</tr>
<tr>
<td>7</td>
<td>1421</td>
<td>7.0</td>
<td>1794</td>
<td>5.2</td>
</tr>
<tr>
<td>8</td>
<td>566</td>
<td>11.4</td>
<td>1106</td>
<td>8.8</td>
</tr>
<tr>
<td>9</td>
<td>419</td>
<td>10.7</td>
<td>904</td>
<td>8.0</td>
</tr>
<tr>
<td>10</td>
<td>236</td>
<td>9.2</td>
<td>747</td>
<td>7.2</td>
</tr>
<tr>
<td>11</td>
<td>192</td>
<td>8.0</td>
<td>619</td>
<td>6.1</td>
</tr>
<tr>
<td>12</td>
<td>82</td>
<td>7.2</td>
<td>359</td>
<td>5.1</td>
</tr>
<tr>
<td>13</td>
<td>67</td>
<td>6.3</td>
<td>327</td>
<td>4.2</td>
</tr>
<tr>
<td>14</td>
<td>53</td>
<td>4.7</td>
<td>284</td>
<td>3.9</td>
</tr>
<tr>
<td>15</td>
<td>20</td>
<td>5.1</td>
<td>211</td>
<td>4.2</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>4.1</td>
<td>178</td>
<td>3.5</td>
</tr>
<tr>
<td>17</td>
<td>7</td>
<td>3.3</td>
<td>140</td>
<td>3.4</td>
</tr>
<tr>
<td>18</td>
<td>6</td>
<td>3.0</td>
<td>116</td>
<td>3.1</td>
</tr>
<tr>
<td>19</td>
<td>2</td>
<td>2.0</td>
<td>76</td>
<td>2.3</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>41</td>
<td>12</td>
<td>1.8</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>12</td>
<td>10</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Table 6 Results for Simulated Data from Class C Compared to Class A and Class B Models
<table>
<thead>
<tr>
<th>Starting Time Frame</th>
<th>Number Rejected From Class A</th>
<th>Number Rejected From Class B</th>
<th>Number Rejected From Class C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>20</td>
<td>17</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>19</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>18</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>6</td>
<td>18</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>18</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>17</td>
<td>18</td>
<td>17</td>
</tr>
<tr>
<td>9</td>
<td>17</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>10</td>
<td>17</td>
<td>19</td>
<td>16</td>
</tr>
<tr>
<td>11</td>
<td>17</td>
<td>18</td>
<td>15</td>
</tr>
<tr>
<td>12</td>
<td>17</td>
<td>18</td>
<td>14</td>
</tr>
<tr>
<td>13</td>
<td>16</td>
<td>18</td>
<td>14</td>
</tr>
<tr>
<td>14</td>
<td>16</td>
<td>18</td>
<td>13</td>
</tr>
<tr>
<td>15</td>
<td>16</td>
<td>17</td>
<td>14</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>17</td>
<td>12</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>17</td>
<td>12</td>
</tr>
<tr>
<td>18</td>
<td>16</td>
<td>17</td>
<td>10</td>
</tr>
<tr>
<td>19</td>
<td>16</td>
<td>17</td>
<td>7</td>
</tr>
<tr>
<td>20</td>
<td>15</td>
<td>16</td>
<td>4</td>
</tr>
<tr>
<td>21</td>
<td>14</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>14</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>23</td>
<td>12</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>24</td>
<td>12</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>25</td>
<td>12</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>26</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>27</td>
<td>10</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>28</td>
<td>10</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>29</td>
<td>10</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>30</td>
<td>8</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>31</td>
<td>8</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>32</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>33</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>34</td>
<td>5</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>35</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 7  False Rejections for Observed Transformed Data Compared to Each Class Model
The performance of the algorithm on comparing Class A and Class B real observations to the model from Class C was consistent with what was observed in the simulation study. Class A and B matches are much different from Class C matches in terms of appearance of plots of their data, so this is not terribly surprising.

In addition to employing the data used in creating the models to assess the performance of the algorithm, we also consider 7 observations that did not belong to any of the modeled classes of safety matches. These 7 observations were rejected as not belonging to Class C after a single observed time frame, regardless of the first observed time frame. Five of the 7 observations were rejected as not belonging to Class B regardless of where the signature began. From these 5 observations, the maximum number of time steps needed to reject the observations was 3 and was rarely more than a single frame of data. The remaining 2 observations did not result in a rejection if the signature begun in the later time steps. However, the observations were rejected from belonging to Class B if the signature was observed early enough in its evolution. Lastly, 2 of the 7 observations were always rejected from belonging to Class A with no more than 2 time frames of data needed for rejection regardless of the starting point. The remaining 5 observations had similar rates to rejection with the exception of the last 2 or 3 time steps in which a rejection was not concluded.

5 Discussion and Future Work

Based on the performance of the discrimination algorithm on simulated data, we have shown that this type of algorithm is potentially useful in determining whether an
incoming data stream is consistent with a model from an important point source event that has previously been characterized in a library. However, the complex nature of the problem suggests that perhaps more data will be needed to adequately model point source events (that will be stored in the library). The models used here were fit based on 20 observations for 2 classes and only 17 for the additional class of data.

In addition to the work discussed here, a small amount of smoothing was applied to the real data before submitting it to the discrimination algorithms to determine if that would improve the algorithm performance characteristics. However, smoothing the data did not result in substantially smaller false rejection rates or quicker times to rejection from the inappropriate class. Therefore, we will not provide further details.

Several aspects of this algorithm could be further investigated before real implementation. First, after looking closely at the values of the cut-points, perhaps one could use a single value for all data classes. This would only be reasonable in the event that the empirically determined cut-points were close in value for a library containing information from many different classes of point sources. Additionally, other methods for determining the cut-points should be investigated to determine which method results in the best optimal characteristics of the discrimination algorithm. For example, perhaps instead of 0.995 quantiles of $MD_j(n_t)$ for all $n_t$, more extreme quantiles should be used when $n_t$ is large.

Next, if such a algorithm is to be employed in real time, a careful investigation might be needed concerning how many wavelengths should be used and also which wavelengths contain the most useful information in terms of making a decision. Minimizing the
amount of information used in computations and stored in the library would make real-time use of this kind of algorithm feasible.

Further, if this kind of algorithm is to be used in real time, after observing some number of frames from an incoming data stream (and potentially before one has observed an “entire” data record), a decision will need to be made. We have not investigated the possibility of introducing a ‘decision’ threshold into the problem. Such a threshold would move along in time and would indicate if one can wait to make a decision or if a decision needs to be made immediately. A motivation for this kind of consideration is a scenario where events are potentially threatening and delay in reacting to a real threat can be disastrous.

The simulation study has shown that the methodology suggested here could be useful. However, the models need to adequately explain the observed data. While refinement is needed, this methodology has potential to be used as a starting point in real-time discrimination between types of point source events.

6 References

Reising, Monica, Max Morris, Stephen Vardeman and Shawn Higbee (2008a) “Modeling Spectral-Temporal Data from Point Source Events”
CHAPTER 3. PRACTICAL CONSIDERATIONS FOR MODELING AND REAL-TIME DISCRIMINATION BASED ON SPECTRAL-TEMPORAL DATA FROM POINT SOURCE EVENTS

Monica Reising, Max Morris, Stephen Vardeman, and Shawn Higbee

1 Abstract

This article discusses important practical considerations regarding the sensor and experimental set-up used in Reising et al. (2008a) as a test bed for modeling efforts with point source spectral-temporal data. Some of these are aspects of the sensor design that may be important considerations in modeling the output of the Chromotomographic Hyperspectral Imaging Spectrometer (CTHIS) sensor or other similar sensors. Experimental choices discussed include the framing rate of the sensor and the fact that all data used when developing the methods of Reising et al. (2008a) and (2008b) were generated by physically stationary point sources of energy. Further, we consider the implications of the fact that while the CTHIS is equipped with a rotating direct vision prism and rotating preprocessing algorithms, these were not utilized in the data collection and modeling in Reising et al. (2008a). Therefore, this article also discusses how data were manually extracted from the focal plane array.
2 Introduction

Reising et al. (2008a) state that in normal use, the Chromotomographic Hyperspectral Imaging Spectrometer (CTHIS) sensor used in data collection for their study makes use of a *rotating prism* and *rotating preprocessing algorithm* that can produce data matrices that are commonly visualized as “waterfall plots.” However, for the purpose of initial development of modeling methods and a discrimination algorithm, the authors used the sensor in fixed-prism mode during data collection and collected raw (i.e. not preprocessed) data from the sensor. This paper will discuss the steps that were then necessary to process these data.

Additionally, we will discuss practical issues that need serious consideration in moving forward with the work presented in Reising et al. (2008a) and Reising et al. (2008b). These include issues related to the specific sensor used as a test case in the present work, possible considerations if a different sensor is of interest, and complications that would result if the point source of energy is in motion.

3 The Sensor

Optical design plays an important role in any sensing problem through the quality of an image or other data it produces. Optical aberrations are of primary concern. These are the faults or defects of the image and are generally measured as departures from what would be expected from the first-order laws of image formulation (The Infrared Handbook 1989). Potential sources of loss of image quality are spherical aberrations, coma, field curvature, distortion, lateral chromatic aberrations, and axial chromatic
aberrations. The optical design of any sensor plays an important role in the quality of image being observed.

The sensor used for data collection by Reising et al. (2008a, 2008b) is a high throughput imaging sensor, the Chromotomographic Hyperspectral Imaging Spectrometer (CTHIS), and is capable of determining spectral components of scenes or objects over a wide dynamic range. The sensor also resolves the time evolution of source spectra for sub-pixel events such as flames, plumes, or flashes (www.solidstatescientific.com). The CTHIS was equipped with a commercially available focal plane array acquired from Santa Barbara Focalplane (www.sbfp.com), but the camera and all the read-out electronics are unique to this sensor.

The CTHIS sensor can be viewed as containing 2 system components, the chromotomographic sensor system and the digital post-processing and image construction system (Dearinger 2004). While the sensor has the capability of constructing hyperspectral images, our interest in the system and digital post-processing is in terms of how sub-pixel events are handled. Dearinger (2004) has particular interest in the system and the hyperspectral imaging. Dearinger (2004) presents a thorough description of the chromotomographic sensor system and provides background on the important components of the CTHIS sensor. The following discussion highlights the need for a thorough understanding of how each sensor component affects the resulting data stream. Each of these sensor components can potentially affect a data stream, and the more that is known about these aspects of the sensor, the more accurately one can model data collected using the sensor.
The chromotomographic sensor system is equipped with a double lens arrangement separated by a field stop aperture (Dearinger 2004). The field stop limits the angular field of view of an optical system (The Infrared Handbook 1989). The second component in the sensor is the prism. This CTHIS sensor has a direct vision prism which is designed for the purpose of dispersing the electromagnetic energy in angular directions with respect to the optic axis. The design of the direct vision prism allows energy at a single wavelength to be passed through the prism to the detector without refraction (Dearinger 2004). Once the energy is passed through the direct vision prism, it comes in contact with the detector. Detector designs are chosen based on the relevant phenomenology, including the relevant portion of the spectrum for a given remote sensing problem. Detector design can be optimized to achieve the highest sensitivity for specific wavelength operation (Jha 2000).

The final component of the chromotomographic sensor system is the focal plane array. Focal plane arrays are chosen based on anticipated signal-to-noise ratios, the implications of possible non-uniformities on the focal plane array, and in some cases the cooling requirements. Further, each focal plane is equipped with a read-out device. Depending on complexity, processing requirements, and cost, the read-out device can use either a charge-coupled device (CCD) detector or a complementary metal-oxide-semiconductor (CMOS) detector (Jha 2000). Magnan (2003) presents the strengths and weakness for both of these types of detectors. When used for photon detection, both are organized as arrays of photodetectors that deliver an electrical signal related to the number of photons that fall on the pixel surface during the integration period. The
sensor used in the modeling presented in Reising et al. (2008) used a CMOS detector.

The data collection of Reising et al. (2008a) was done at a framing rate of 80 frames per second. We assume, but have not verified, that the change in the characteristic event evolution is smooth from one frame of data to the next at this framing rate. It might be of interest to study whether the characteristic evolution of a class of events appears to be different when the framing rate is different. If a faster framing rate reveals subtle changes in the characteristic evolution of the event that were not observable at a framing rate of 80 frames per second, would event discrimination still be effective at the lower framing rate if the timing were slightly off in observing the incoming important event?

On the other hand, another natural extention to the work in Reising et al. (2008a, 2008b) would be studying data collected at a slower framing rate (less temporal resolution). A starting point for such an investigation is to simply add the output (read-out voltages) at each pixel on the focal plane array across some number of time frames. This seems a reasonable representation of what the observed data would be for the event observed at a slower framing rate, assuming there is no dead time between consecutive time frames. The resulting counts in the raw data would be larger, but the same modeling ideas presented in Reising et al. (2008a) should be applicable and the existing data provide a potential test bed for comparing modeling and discrimination effectiveness at several rates.

This kind of analysis would be important in a situation where data collected to develop a library of event characterizations for later use in a discrimination routine are taken with the camera set at a faster framing rate than what would be used in real
time. There are some additional issues that need consideration in this context, however. Suppose, for example, that data were collected at a framing rate 4 times faster than the one used in practice. One would have 4 times as many time frames in the training data as in the record on an event of the same length. Roughly speaking, one could simply add responses in sets of 4 consecutive time periods and use the resulting data to develop a model for an event class. The model developed from the summed data could then in turn be used in the discrimination algorithm. However, this line of development raises the issue of time registration. After some form of temporal alignment of training data collected at the higher framing rate, there are still 4 possible ways in which the data could be summed across time frames. For instance, one could add data from time frames 1 to 4, 5 to 8, etc. Additionally, one could also add data from time frames 2 to 5, 6 to 9, etc. The manner in which the summing is done will presumably affect the model that is ultimately used in the discrimination.

In light of this potentially arbitrary element the issue of registration introduces into modeling and any associated discrimination algorithm, it seems clear that use of a single summed version of a training data record is inadequate. It is a reasonable assumption that real events of interest have synchronizations that are “uniformly distributed” relative to both the rapid framing of the sensor in the collection of training data and the slower framing in real-time discrimination. Under this assumption, a plausible method of generating training data for modeling at the lower framing rate is to create and use all four possible summed versions of an original training data record.

An alternative to modeling and discriminating based on summing data over time and
creating a model for the summed counts would be to operate on the incoming data and work with the model that has been stored in the library. Suppose that an incoming time frame of data corresponds to 2 consecutive time frames at the training data framing rate represented in the library. That is let $Y$ represent the incoming data and $Y = X_1 + X_2$ where $X_1$ and $X_2$ are raw counts obtained from the sensor at the faster framing rate. In the modeing presented in Reising et al. (2008a), $X_1$ and $X_2$ were transformed according to

$$h(\theta) = \begin{cases} 
\ln(\theta) & \text{for } \theta \geq 2 \\
\theta(\ln(2)/2) & \text{for } -2 < \theta < 2 \\
-\ln(|\theta|) & \text{for } \theta \leq -2 
\end{cases}$$  \hspace{1cm} (1)$$

before means and covariances were fit.

Since $h$ is nonlinear, means and covariance matrices for $Y$ are not simple functions of those for $(X_1, X_2)$. However, “delta-method” approximations might be adequate, for constructing approximate slow-rate reference libraries from characterizations made at higher framing rates.

The training data used in Reising et al. (2008a, 2008b) could be used to determine which of these approaches might be more useful. More stored class comparisons would need to be made, however, “on the fly” computing based on data from the library would not be needed. If these ideas were taken forward to be used in real time, a study could be conducted to determine which of these possible solutions would be more feasible.

A situation might also arise (perhaps as a result of advances in sensor capability and computing power) in which data from a sensor with framing rate faster than that of the sensor used to create an event library are available. One might, for instance,
be observing an event at twice the temporal resolution that was used when creating
the means and covariance structures for an event library. In this case, counts in the
input stream could be summed and the transformation used in Reising et al. (2008a)
applied to the summed counts. If there is no appreciable dead time in photon collection
of the sensor between consecutive frames this should produce observations legitimately
comparable to the existing library. One could use the training data in Reising et al.
(2008a) to determine how feasible this would be by summing consecutive time frames
of data, building models for the data with less temporal resolution, and then using the
original (raw) data with the discrimination algorithm.

One might also consider summing read-out voltages across pixels on the focal plane
array as a means of investigating reduced spectral resolution. A model could be de-
veloped where information was combined across the pixels (wavelength indices). This
could also be employed if the dimensionality of the matrices stored in the library are too
big to handle from a computation or data storage standpoint. This would allow for the
retention of more spectral information to be used in the discrimination algorithm rather
than eliminating spectral bands. It is clear how this would be applied to our training
data as the prism was not spinning and the observed counts for each wavelength index
were obtained from a single pixel across all time frames. However, if one were to extend
this work to data output from the normal “spinning prism” processing algorithms of
the sensor, the extension might not be as straightforward. A clear understanding of the
processing algorithms would be needed to determine the most appropriate way to move
forward in terms of combining information across wavelengths.
A further extension of this work may be to data collected from a completely different type of sensor. There are many different ways to sample an event; spatially, spectrally and temporally, and CTHIS sensing is only one possible way. It would be of interest to know if the modeling techniques presented in Reising et al. (2008a) could be applied to data taken from a completely different type of sensor.

4 Raw Data Extraction

Reising et al. (2008a) preprocessed data manually because the rotating prisms and preprocessing algorithms used in normal sensor operation were disabled. As was mentioned earlier, energy at a single wavelength, called the undeviated wavelength, is passed through the prisms unrefracted. Energy at other wavelengths is dispersed along what then appears as a bright “line” on the focal plane according to the refractive properties and orientation of the prism. In standard operation of the sensor, the prism is rotated continuously so that the bright line appears to rotate about the location on the focal plane corresponding to the undeviated wavelength. In this standard operation, “background intensity” based on a set of reference frames taken prior to event initiation is subtracted from each data frame. For each prism orientation, a different algorithm is applied to infer intensities at various wavelengths from read-out voltages that are functions of photon counts in pixels on the focal plane near an appropriate line running through the discrete grid of pixels (Pellegrini et al. 2004). After a series of frames are collected, an algorithm is used to extract a spectral image from these data (Mooney et al. 1997).

In its standard mode of usage, this sensor uses pseudo-imaging to provide spectral-
temporal signatures and the locations of rapidly changing events within a given field of view. Pseudo-imaging is the process of taking data obtained from the focal plane array and creating a single spectral profile and using the spectral profiles from multiple “snap-shots” to create a spectral-temporal signature from an observed event (Weeks et al. 2004). To extract the spectral content from intensities read from the 2-dimensional focal plane array, a technique based on radial basis functions is employed (Nelson 2004). The sensor has a direct vision prism that spreads energy from a point source onto the focal plane array in a line (Pellegrini et al. 2004). Then, to extract a single spectrum, for each fixed time frame of data, brightness values are estimated as a linear combination of values of radial basis functions placed along the bright line (Nelson 2004). These are used to obtain intensity values that take into account the fact that the grid of pixels on the focal plane array are not oriented along the line of refraction of the beam from the point source, except possibly when the prism is positioned to spread light parallel to the rows or columns of pixels on the array.

Since the data of Reising et al. (2008a) were collected with the direct vision prism stationary, raw data were extracted “manually” (without the use of the standard processing software). The raw data collected consisted of series of focal plane gray-scale images like that in Figure 1. Specifically, the prism orientation was fixed and no “automatic” background subtraction was performed by the sensor. For a given experimental set-up (a sensor position relative to a pin-hole through which burning matches were viewed) the spatial location of the undeviated wavelength on the focal plane was determined with the use of a blackbody (a calibrated radiation source). A single observation was taken from
the blackbody at both a vertical and horizontal prism alignment. The pixel on the focal plane where the bright lines in these observations intersect corresponds to the undeviated wavelength (see Figure 2) in either prism orientation. This spatial-spectral calibration was performed prior to the initiation of data collection and was repeated several times during data collection. Prior to recording data for a new “event” class, this alignment was repeated while holding the experimental features constant. Each experimental series of gray-scale images was made with the same prism orientation (undisturbed from the pre-collection exercise) so the pixel location corresponding to the undeviated wavelength is known for every series of images.

After all data were collected, the column of pixels containing the bright line in each
experimental gray-scale image was manually extracted. The intensities read from each pixel in the bright column, over repeated sample times, were compiled to create a matrix of data. Each row in this matrix corresponded to a single frame of data and each column to a single wavelength. The collection of all these frames of data are used to make what are commonly referred to as “waterfall plots.”

5 Intensity Shift Parameter

The modeling of spectral-temporal data presented in Reising et al. (2008a) includes estimation of an intensity shift parameter for each observed event. If two identical events are observed through the same atmosphere, but one event occurs closer to the
sensor, the sensor will produce higher intensity values for the nearer event than for the event further from the sensor. This drives the need for the intensity shift parameter in the model. This intensity shift was dealt with using a difference in the average intensity value (across time and wavelength) between the transformed observation and the mean function for a particular class. However, the data used in modeling in Reising et al. (2008a) were collected under controlled experimental conditions where the distance between the sensor and the object observed remained virtually unchanged. Hence, while the characteristics of a particular event class that were stored in the library to be used in discrimination in Reising et al. (2008b) were obtained from the data collected under these controlled experimental conditions these stored characteristics are appropriate for discrimination for events observed from any fixed distance.

But suppose one wanted to implement the modeling ideas and the discrimination algorithm in real-time for events that were not physically stationary. Data would need to be obtained from enough moving events within a class to determine how the mean function and spectral-temporal covariance structure might be affected over the time required to record an event, and one would need to determine the extent to which the modeling techniques presented in Reising et al. (2008a) would need to be generalized for modeling for a class of events where the individual point sources are moving. The mean function would have to describe not only the characteristic evolution for physically stationary events within the class, but would also have to adequately represent events in which the distance from the sensor evolves in time as well. It seems at this point that the manner in which the intensity shift parameter was handled in Reising et al. (2008a)
would not be adequate under a scenario where the point source events are moving. The intensity shift parameter was applied to all intensity values for a given observation. However, if an observed heat source was moving relative to the sensor, it seems that the distance-induced change in overall intensity would have to be handled in some other fashion, so that intensity correction can vary over time.

Only the beginning steps have been taken towards developing a discrimination algorithm and using it in real-time for event discrimination. There are many features of the sensor and the experimental conditions that would need further consideration for the modeling ideas of Reising et al. (2008a, 2008b) to be more broadly applicable.

6 References


Reising, Monica, Max Morris, Stephen Vardeman and Shawn Higbee (2008a) “Modeling Spectral-Temporal Data from Point Source Events”

Reising, Monica, Max Morris, Stephen Vardeman and Shawn Higbee (2008b) “An Algorithm for Real-Time Discrimination Based on Spectral-Temporal Data from Point Source Events”

*The Infrared Handbook* (1989) Printed in Cooperation with *SPIE - The International Society for Optical Engineering*

CONCLUSIONS AND FUTURE WORK

1 Conclusions

This work has described a methodology for modeling spectral-temporal data from point source events. The hope is that the ideas presented would be applicable to similar data collected from other types of sensors or from this sensor where data was collected at a different framing rate. Further, initial stages have been taken to describe an algorithm that could be used in real-time to determine if an incoming data stream is consistent with a class of energetic events that has previously been modeled and has characteristics stored in a library.

This work was based on a modified real sensor in which the spinning prism and pre-processing algorithms were not used. Much of this work should be described as “proof of concept,” as this sensor would not be used in practice without spinning prisms or pre-processing algorithms. This work would represent preliminary steps toward developing a discrimination algorithm for use in real time.

2 Future Research

In the most immediate future, the models and methods for determining the cut-points in the discrimination algorithm would need further scrutiny. As it stands, the
algorithm did not produce controlled error rates when the observed data was used. The algorithm nearly always falsely rejected an incoming data stream as consistent with the class of energetic events to which it belongs, especially for data streams that begin early in the evolution of an event. This could be rectified by further refining the method for determining the cut-points. Further, it is possible that the present model simply does not adequately described the data from each class of events for this purpose. Future work would include taking steps necessary so that the discrimination algorithm would perform acceptably in terms of false rejections and time to rejection for the safety match data.

In the more distant future, data could be obtained and modeled from the sensor in standard operating mode. Further, data from different sensors could be obtained to determine if these techniques apply more broadly. Finally, one could determine how, if possible, data and library information for important energetic events could be used across different sensors.