



**U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND
CHEMICAL BIOLOGICAL CENTER**

ABERDEEN PROVING GROUND, MD 21010-5424

DEVCOM CBC-TR-1810

**Explosive Forensic Technology:
Hyperspectral Real-Time Threat Anomaly
Detection (Hyper Thread)**

Eric R. Languirand

Darren K. Emge

Justin M. Curtiss

RESEARCH AND TECHNOLOGY DIRECTORATE

July 2022

Disclaimer

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorizing documents.

REPORT DOCUMENTATION PAGE				<i>Form Approved</i> OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 h per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) XX-07-2022		2. REPORT TYPE Final		3. DATES COVERED (From - To) Mar 2015 – Sep 2021	
4. TITLE AND SUBTITLE Explosive Forensic Technology: Hyperspectral Real-Time Threat Anomaly Detection (Hyper Thread)				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER 0602144	
6. AUTHOR(S) Languirand, Eric R.; Emge, Darren K.; Curtiss, Justin M.				5d. PROJECT NUMBER BL2	
				5e. TASK NUMBER T01	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Director, DEVCOM CBC, ATTN: FCDD-CBR-CP, APG, MD 21010-5424				8. PERFORMING ORGANIZATION REPORT NUMBER DEVCOM CBC-TR-1810	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Combat Capabilities Development Command Chemical Biological Center; 5183 Blackhawk Road, Aberdeen Proving Ground, MD 21010-5424				10. SPONSOR/MONITOR'S ACRONYM(S) DEVCOM CBC	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release: distribution unlimited.					
13. SUPPLEMENTARY NOTES U.S. Army Combat Capabilities Development Command Chemical Biological Center (DEVCOM CBC) was previously known as U.S. Army Edgewood Chemical Biological Center (ECBC).					
14. ABSTRACT: (Limit 200 words) Hyperspectral line scanners provide a wealth of data, from which information can be derived and potential threats can be realized. However, real-time analysis of this data is difficult due to the sheer volume of data that must be processed; therefore, this data has traditionally been post-processed. We used statistical representations of the incoming data by looking at higher-order statistics (skewness and kurtosis) and information theory (entropy) to provide probability distribution function-specific data for each of the incoming spectra, thereby reducing the computational burden. In this work from fiscal year 2020–2021, we show that our statistical representations of the data can be used for anomaly detection. We did this through collection of data, treatment of experimental and simulated spectra, ground-truth development for statistical analysis, and an analysis into the use of pretreatment with our data. Furthermore, we determined that implementation of our algorithm using semi-supervised machine learning results in real-time analysis (100 ms frame rate, 250 spectra per frame) of the hyperspectral data we obtain. This algorithm can be implemented in a scenario when immediate situational awareness is necessary, thereby increasing Warfighter lethality.					
15. SUBJECT TERMS					
Anomaly detection		Surface threat		Hyperspectral	
Ground truth		Skewness		Kurtosis	
				Machine learning	
				Entropy	
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 58	19a. NAME OF RESPONSIBLE PERSON Renu B. Rastogi
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U			19b. TELEPHONE NUMBER (include area code) (410) 436-7545

Blank

PREFACE

The work described in this report was authorized under project no. 0602144/BL2. The work was started in March 2015 and completed in September 2021. At the time this work was performed, the U.S. Army Combat Capabilities Development Command Chemical Biological Center (DEVCOM CBC; Aberdeen Proving Ground, MD) was known as the U.S. Army Edgewood Chemical Biological Center (ECBC).

The use of either trade or manufacturers' names in this report does not constitute an official endorsement of any commercial products. This report may not be cited for purposes of advertisement.

The text of this report is published as received and was not edited by the Technical Releases Office, DEVCOM CBC.

This report has been approved for public release.

Acknowledgments

The authors acknowledge the following individuals for their hard work and assistance with the execution of this technical program:

- Dr. Steven Kay (University of Rhode Island; Kingston, RI) for his work in providing mathematical expertise for anomaly detection;
- Mr. Ian Emge (SciTech; Havre de Grace, MD) for his work in transitioning the MATLAB code to Python; and
- Mr. Kevin Hung (DEVCOM CBC Spectroscopy Branch) for his work in writing the software required for real-time analysis.

Blank

CONTENTS

	PREFACE	iii
1.	INTRODUCTION	1
2.	MATERIALS AND METHODS.....	5
2.1	Materials Used	5
2.2	Instrumentation and Infrastructure.....	6
2.3	Analysis Methods.....	7
3.	RESULTS AND DISCUSSION	8
3.1	Understanding How Distributions Change HOS and Entropy	8
3.1.1.1	Kolmogorov-Smirnov Test for Normality	8
3.1.1.2	Effects of Bimodal Transformation on HOS and Entropy.....	9
3.2	Kurtosis As a Threshold.....	14
3.2.1	Otsu Threshold for Kurtosis.....	14
3.2.1.1	Simulated Data.....	14
3.2.1.2	Laboratory Data	16
3.3	Parameterization of Spectral Data	19
3.4	Real Time Algorithm Development.....	21
3.4.1	Anomaly Algorithm	22
3.4.1.1	Background Determination	22
3.4.1.1.1	Analysis of Training Data.....	22
3.4.1.1.2	Determination of An Anomaly	23
3.4.1.2	Real Time Application.....	24
3.4.1.2.1	Real Time Simulator	24
3.4.1.2.2	Real Time Hyperspectral Line Scan Analysis	26
3.5	Ground Truth Analysis	29
3.5.1	Synthetic Spectra for the Ground Truth.....	30
3.5.2	Ground Truth Performance	32
3.5.2.1	Epsilon Optimization	33
3.5.3	Non-Uniformity in Ground Truth Mask	34
3.6	Correction and Illumination Analysis.....	36
3.6.1	Radiometric Corrections	36
3.6.2	Illumination Analysis.....	37
3.7	Improvements Through Programing Languages	38
4.	CONCLUSIONS	39
	LITERATURE CITED	41
	ACRONYMS AND ABBREVIATIONS	45

FIGURES

1.	A hyperspectral hypercube consisting of spatially separated spectra (left) which can be represented as a single frame of scan of spectra (middle) and can be processed as a probability distribution resulting in parameters that can be used for analysis (right).	2
2.	A signal (e.g., spectrum) (A) is binned via the intensity value on the y-axis of the spectrum and counted as number of occurrences (B). This distribution can then be utilized for information theoretic calculations or fitted by an appropriate function for higher-order statistics (C).	2
3.	Estimated distributions from changing the HOS (A), how the HOS change, as represented in the center of (A) with a normal distribution in the center (B) and the values that the HOS take on with a normal distribution (C).	4
4.	Custom built rack for indoor and outdoor testing from the side (left) and from the end of the rack system (right). The Co-Aligned VNIR-SWIR is mounted on a stage across the length of the top of the rack (A) and is controlled via hydraulic pistons to adjust the height of the rack system (B). The surfaces that are analyzed are artificially illuminated with halogen lights (C).	7
5.	A contour line plot of KS statistic as a function of changing skewness and kurtosis (A) and a two-dimensional representation of A (B). The red star represents the KS statistic of a normal distribution.	9
6.:	The simulated distributions shifting and separating via an increasing second mode mean (A) and the effects of that separation on the skewness (B), kurtosis (C), entropy (D), and KS statistic (E).	11
7.	Simulated distributions in which a single distribution has a second mode of the distribution growing while the first mode is reduced (A) and the effects this has on skewness (B), kurtosis (C), entropy (D), and the KS statistic (E).	13
8.	Representative data from two sets with 0% data in the second population (A, top) and 3.5% data in the second population (A, bottom) and selected representative histograms with the red bar representing the Otsu threshold (B). The Otsu threshold for 100 trials for up to 50% of data in the second population (C).	16
9.	Laboratory collected data for kurtosis of aged asphalt (A, blue) and asphalt road (A, red) from 10,000 representative spectra. The Otsu threshold (*) from spectrally mixing the two spectra with the average kurtosis +/- the standard deviation represented by the solid and dashed lines, respectively for each set (B). The representative histograms of the Otsu threshold with the kurtosis data (C).	18
10.	A representative image of the regions of interest identified for each analyte (A) and the parameterizing of the spectra within those regions for skewness, kurtosis and entropy shown in a three-dimensional scatter plot (B).	19
11.	Parallel coordinates of ground materials in six dimensions and a zoomed in area to show differentiability of ammonia nitrate (blue circles) and potassium chlorate (green circles) (A). Parallel coordinates of surface materials highlighting ammonia nitrate and potassium chlorate (B).	21
12.	A block diagram of the algorithm used within this work (A) and pseudo code of the algorithm used within this work (B).	23

13.	Pseudo code for RTS with two instances of MATLAB. The first instance (left) loads a hypercube and creates a data stream to be processed (right).....	25
14.	The data stream produced by the RTS showing the memory writing speed and the signal stream speed (A) and the timings associated with processing the training data/background (B, left) and the test area (B, right) with a red-dashed line representing the 0.1s threshold goal we had.	26
15.	Real time operation of the instrument with the anomaly detection algorithm employed for 150 line scans with respect to the target timing (red dashed line) (A), and a long-term collection of 2500 lines and the coinciding approximate amount of data processed within that time (B).	28
16.	Labeled representation of real time anomaly detection results (A), the SWIR anomaly detections (red) and areas considered not an anomaly (green) (B), and VNIR anomaly detection (red) and areas considered not an anomaly (green) (C).....	29
17.	A binarized mask of the developed ground truth data set with both alphanumeric descriptions and square areas of where an anomaly is present (A) and a magnified view of various aspects of the ground truth (B).....	30
18.	Simulated dissemination patterns of 1, 5, 10, and 25g/m ² (A) and the correlated area fraction of drops as a function of mass density (B).....	31
19.	Binarized ground truth mask (A), a scaled mask showing the concentration gradient that is realized when the simulated spectra are applied (B), and the results of the anomaly detection algorithm showing a LOAD of 5g/m ² (C).....	33
20.	Simulated dissemination and a representation of the pixel area during travel at 5mph (red boxes, A), the percentage of pixels with droplets residing within the integrated area as a function of ground concentration (B), the uniform coverage of 0.1, 0.5, and 1g/m ² , and the resulting coverage with non-uniform coverage.	35
21.	Temperature (black points) and humidity (red points) levels during the outdoor data acquisition.	36
22.	Differences in anomaly detections between implementing our anomaly in MATLAB and Python. A difference map showing the pixels that were anomalous with implementation of Python in pink or MATLAB in yellow (A), common pixels both identified as anomalies in both Python and MATLAB (B), and the speed improvement associated with implementation in Python (C).....	39

TABLES

1.	Materials and Data Used in This Report.....	5
2.	Change in LOAD With Radiometric Corrections.....	37
3.	Change in LOAD with Solar Illumination.....	38

Blank

Explosive Forensic Technology: Hyperspectral Real-Time Threat Anomaly Detection (Hyper Thread)

1. INTRODUCTION

Gaining a tactical edge does not always mean decisively determining information about ones surroundings. One way a tactical edge can be gained is through anomaly detection, e.g., finding things that are unique or not expected. According to Chandola et al., anomaly detection is “finding patterns in data that do not conform to expected behavior”.¹ Gaining a tactical edge can be as simple as understanding when something doesn’t fit a pattern and could be of concern. From the perspective of a Warfighter gaining a tactical edge in theater, anomalous detections can be indicative of targets or situations of interest from a noisy signal stream. Therefore, anomaly detection can loosely be viewed through the eyes of signal detection theory in terms of a noisy signal from which a decision must be made.²⁻³ More specifically, anomaly detection can be a type of novelty detection whereby the novelty is not part of the training methods.⁴⁻⁵ Many techniques for anomaly detection exist, and these include (but are not limited to) classifications, clustering, nearest neighbor, statistical, information theory, and spectral anomaly detection techniques.¹

The type of anomaly detector employed doesn’t just depend on the type of data that is being analyzed, but more so on how the data is managed or processed. Hyperspectral data, for example, is spatial information that is collected simultaneously with spectral information, creating a three-dimensional hypercube of information, depicted in Figure 1 (left) and can be useful for surface threat analysis. There are also techniques that can find anomalies in spectral data.⁶⁻⁷ However, they are often computationally expensive when applied to hyperspectral data. Most hyperspectral processing techniques process the entire hypercube at once.⁸⁻¹⁰ Processing data in real time can become computationally prohibitive due to the sheer volume of data that needs to be sorted and processed in real time and methods to reduce computational burden must be realized to approach real time analysis.

There are methods that aim to reduce the amount of data that is processed in hyperspectral analysis and these mostly rely on multiband analysis.^{8, 10-13} Several spectral bands can be utilized to gain pertinent information from the hyperspectral dataset and is represented in Figure 1 (middle and right). This can effectively reduce the amount of data that is being processed, but it is also reliant on a small portion of the spectral data present resulting in a more

rapid process.

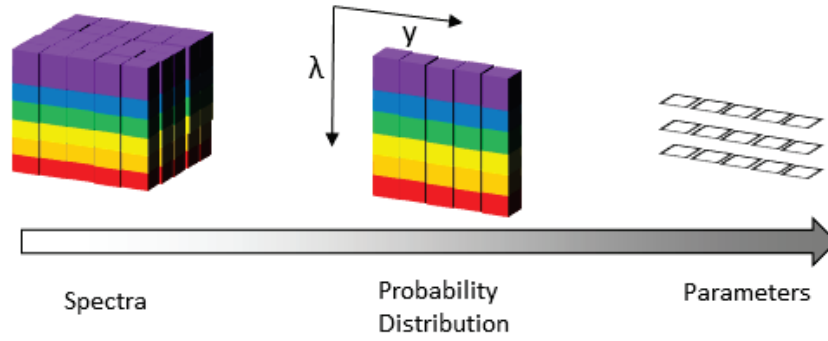


Figure 1: A hyperspectral hypercube consisting of spatially separated spectra (left) which can be represented as a single frame of scan of spectra (middle) and can be processed as a probability distribution resulting in parameters that can be used for analysis (right).

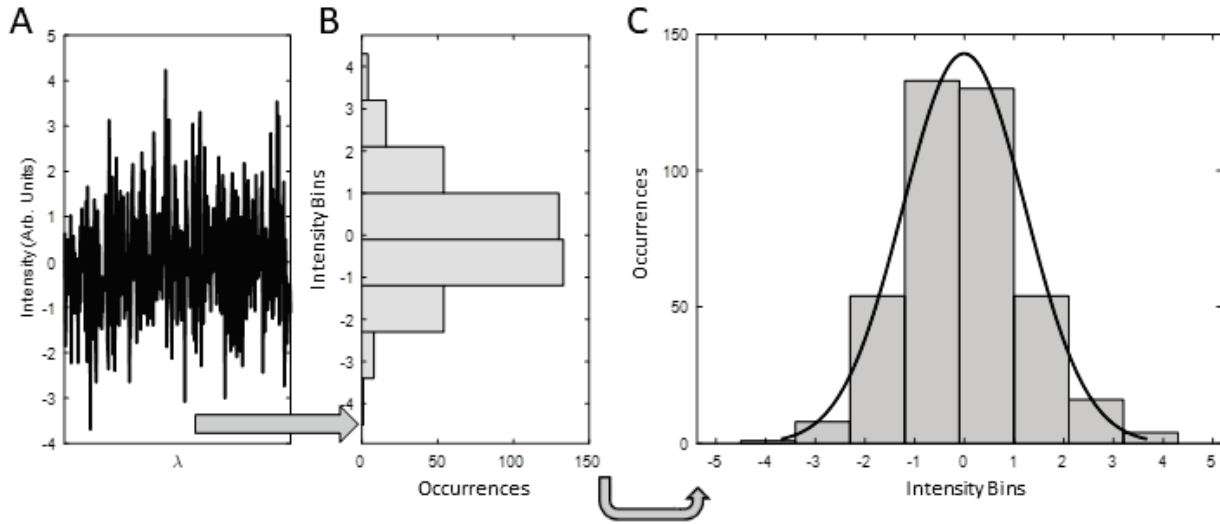


Figure 2: A signal (e.g., spectrum) (A) is binned via the intensity value on the y-axis of the spectrum and counted as number of occurrences (B). This distribution can then be utilized for information theoretic calculations or fitted by an appropriate function for higher-order statistics (C).

Two methods that we employ to reduce the dimensionality of a signal, first is to examine the probability density function (PDF) of the spectral values and calculate the higher order statistics (HOS)¹⁴⁻¹⁷ and information theoretic techniques.¹⁸⁻²¹ The PDF of signals (Figure 2B and Figure 2A, respectively), including those of spectra, are produced by grouping the intensity values into bins and counting the occurrences. When binned and counted, the PDF can be used to calculate information theoretic parameters or can be fitted with an appropriate function to calculate the associated HOS, as illustrated in Figure 2C.

Mathematically, HOS are a summation series and are not computationally burdensome thereby decreasing the time it takes to process the data. Specifically, in this work we seek to employ skewness and kurtosis which are the third and fourth moments of the PDF, respectively. HOS are described mathematically as:

$$m_k = \frac{1}{n\sigma^k} \sum_{i=1}^n (x_i - \bar{x})^k \quad (1)$$

Where m_k is the k^{th} moment, n is the length of data, σ is the standard deviation of the data, $\bar{x}$ is the sample mean of the data. Skewness provides $k = 3$ and kurtosis provides $k = 4$. A visual representation of HOS is in Figure 3. The way in which the distribution changes as the skewness and kurtosis changes is displayed in Figure 3A. The recessed gray arrows in the center of Figure 3A are represented in Figure 3B. It can be clearly seen that as skewness becomes more positive (upper row) the distribution shifts to the left whereas it shifts to the right as the skewness is more negative (bottom row). As kurtosis becomes greater than 3 (right column), the distributions becomes more pointed whereas the distribution becomes more flat-topped when kurtosis is less than 3 (left column). A normal distribution has a skewness and kurtosis equal to 0 and 3 respectively, as indicated by the middle distribution in Figure 3A and shown in Figure 3C. Therefore, fitting a PDF speaks to the shape of the PDF and when the PDF changes due to spectral features, the associated HOS will also change.

Entropy utilizes a logarithmic function, which can be quickly and easily determine using lookup tables, and is also a summation:

$$S = - \sum_i^n p(x_i) \log(p(x_i)) \quad (2)$$

Where S is entropy, and p is probability of event x happening and is determined through normalized occurrences. While the HOS help explain the shape of the PDF, the entropy speaks to the amount of information that is available in a particular PDF. This will change as the number (but not position) of peaks change in the spectrum as this necessarily changes the PDF. A combination of these can be used to describe spectral changes.²²⁻²³

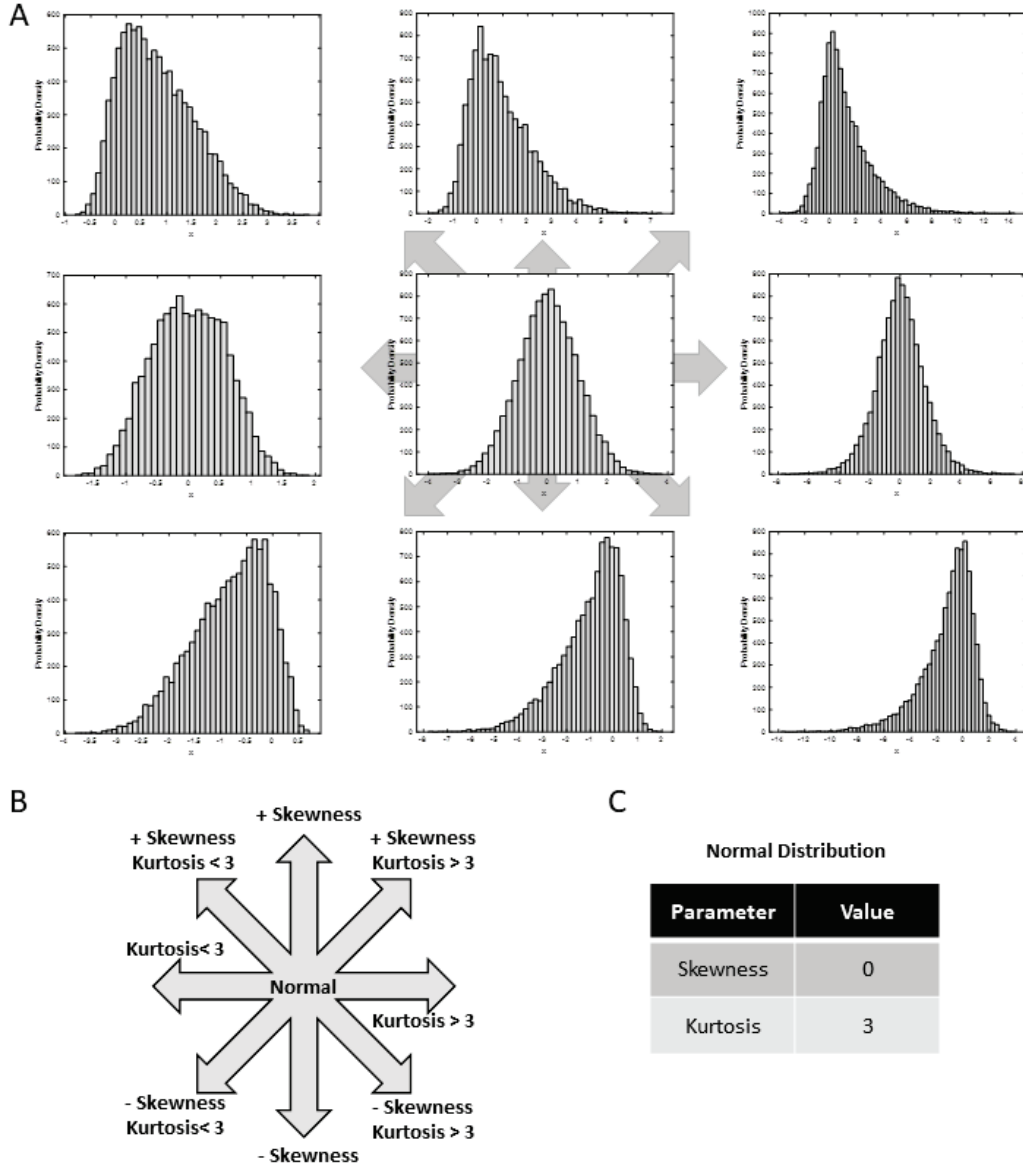


Figure 3: Estimated distributions from changing the HOS (A), how the HOS change, as represented in the center of (A) with a normal distribution in the center (B) and the values that the HOS take on with a normal distribution (C).

In this report we provide our research completed over fiscal years 2020 and 2021 towards hyperspectral real-time threat anomaly detection (Hyper Thread) with a focus on explosives. First, we show how different PDFs change entropy and HOS including when more than one mode is present in the PDF. This is followed with an analysis of how kurtosis may potentially be used as mechanism for identification of a threshold. Section 3.3 delves into the parameterization of hyperspectral data and how the data is used from the instrumentation that we employed. This is followed by an outline of the algorithm we developed and how the above parameters are employed in real time analysis, along with full implementation of the algorithm in a laboratory environment in Section 3.4. The development and analysis of a ground truth to appropriately describe the performance of the algorithm is discussed in Section 3.5. Finally, in

Section 3.6 the need to use radiometric corrections and an analysis into differences in solar and artificial illumination are discussed.

2. MATERIALS AND METHODS

2.1 Materials Used

The materials used within this report are outlined in Table 1 and describe the material, the abbreviation used in this report, the supplier and company information for the materials, the illumination type, and the spectral region in which the data were collected and used. All materials were purchased or obtained from the supplier listed in Table 1 and used as received.

Table 1: Materials and Data Used in This Report

Material (Abbreviation)	Supplier	Artificial Illumination	Solar Illumination	VNIR ^e	SWIR ^f
Acetaminophen (ACE)	Mallinckrodt ^a	✓	✓	✓	✓
Aged Asphalt (AA)	Aerico ^b		✓	✓	✓
Ammonia Nitrate (AN)	Spectroscopy Branch ^c	✓	✓	✓	✓
Asphalt (AS)	Aerico ^b	✓	✓	✓	✓
Black Car Panel (BCP)	Spectroscopy Branch ^c	✓	✓	✓	✓
Brushed Concrete (BC)	Aerico ^b	✓	✓	✓	✓
Glucosamine (GA)	Headwall Photonics ^d	✓			✓
Grass (GR)	Headwall Photonics ^d		✓	✓	✓
Gravel (G)	Aerico ^b	✓	✓	✓	✓
Parking Lot (PL)	Headwall Photonics ^d	✓	✓	✓	✓

Potassium Chlorate (PC)	Spectroscopy Branch ^c	✓	✓	✓	✓
Red Car Panel (RCP)	Spectroscopy Branch ^c	✓	✓	✓	✓
Red Dirt (RD)	Aerico ^b	✓	✓	✓	✓
Rusted Metal (RM)	Spectroscopy Branch ^c	✓	✓	✓	✓
Smooth Concrete (SC)	Aerico ^b	✓		✓	✓
Sodium Bicarbonate (SBC)	Headwall Photonics ^d	✓			✓
Sodium Chloride (SC)	Headwall Photonics ^d	✓			✓
Sucrose (S)	Headwall Photonics ^d	✓			✓
Truvia (T)	Headwall Photonics ^d	✓			✓
White Car Panel (WCP)	Spectroscopy Branch ^c	✓	✓	✓	✓
White Clay (WC)	Headwall Photonics ^d	✓			✓

^a Mallinckrodt Pharmaceuticals, Staines-Upon-Thames, Surrey, UK

^b Aerico Industries Incorporated, Forest Hill, MD, USA

^c DEVCOM CBC Spectroscopy Branch, Aberdeen Proving Ground, MD, USA

^d Headwall Photonics, Inc., Bolton, MA, USA

^e Data obtained in the visible to near infrared region of the electromagnetic spectrum

^f Data obtained in the shortwave infrared region of the electromagnetic spectrum

2.2 Instrumentation and Infrastructure

Data were collected for PL and GR at Headwall Photonics headquarters in Bolton, MA whereby a manufacture-owned Co-Aligned visible and near-infrared-short-wave infrared (VNIR-SWIR) (Headwall Incorporated, Bolton, MA) was mounted to a DJIM600 Pro unmanned aerial system (UAS). A flight pattern was identified over a parking lot and grassy area and the data were collected from between 5 and 30 mph. The GA, SBC, SC, S, T, and WC was data obtained by Headwall Photonics with a SWIR sensor equivalent to the SWIR sensor in the Co-Aligned VNIR-SWIR instrument and provided to DEVCOM CBC personnel.

A hyperspectral line scanner (Co-Aligned VNIR-SWIR) was used in the acquisition of all data obtained at DEVCOM CBC. Data was obtained by the Co-Aligned VNIR-SWIR both indoor and outdoor using a custom built rack system equipped with hydraulic lifts for mobility. The rack, shown Figure 4 was built to raise and lower with the hydraulic pistons such that the sensor could be secured 2m from the surface (defined by interior optics of the line scanner) and be transported outside. The instrument is secured to a 3m long linear stage (Zaber Technologies) and was translated at 50mm/sec when data was obtained. Signal was optimized based on lighting conditions and surface reflection prior to obtaining data.

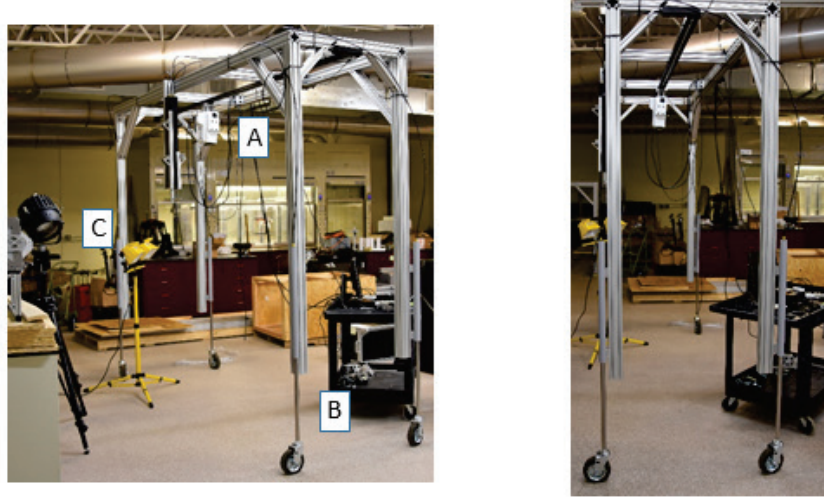


Figure 4: Custom built rack for indoor and outdoor testing from the side (left) and from the end of the rack system (right). The Co-Aligned VNIR-SWIR is mounted on a stage across the length of the top of the rack (A) and is controlled via hydraulic pistons to adjust the height of the rack system (B). The surfaces that are analyzed are artificially illuminated with halogen lights (C).

2.3 Analysis Methods

All post-processed data was acquired with the manufacture's software, Hypersec III[®]. Real time data acquisitions and all processing of the hyperspectral data was completed with MATLAB R2019b. Converting the MATLAB code into Python code was completed with the Anaconda Navigator and the Spyder 4.1.4 integrated development environment was used (see Section 3.7).

All simulated distribution such as those shown in Section 3.1 were simulated using sinh-arcsinh distributions which provide a random variable with defined skewness (ϵ) and kurtosis (δ). The properties of this family of distributions is found in detail elsewhere.²⁴⁻²⁶ In short, a random number can be generated with a sinh-arcsinh probability distribution function and is defined as:

$$X_{\mu,\sigma,\epsilon,\delta} = \sinh[\delta^{-1}\{\sinh^{-1}(Z_{\mu,\sigma}) - \epsilon\}] \quad (3)$$

Where $X_{\mu,\sigma,\epsilon,\delta}$ is a random number generated from a distribution with skewness and kurtosis from a randomly generated number of a normal distribution with a mean of μ and a standard deviation of σ . Sinh-arcsinh distributions were generated with a custom written

MATLAB script that generated at least 1000 normal random numbers that were then used to generate random numbers from a user defined mean, standard deviation, skewness, and kurtosis through Equation 3.

3. RESULTS AND DISCUSSION

3.1 Understanding How Distributions Change HOS and Entropy

Understanding how the parameters that we employed behave is important and the first step in understanding how to work with the data. As such, we can evoke the Central Limit Theorem (CLT) stating that as you increase the sample size of an independent random variable, their normalized sum tends towards a normal (bell-curve shaped) distribution.²⁷⁻²⁸ The HOS and entropy can be considered an independent random variable for each particular PDF of a spectrum (e.g., Figure 3). Mathematically, the HOS and entropy are defined specifically by the PDF, and are therefore independent of other PDFs, as shown in Equation 1 and Equation 2. That is, a specific analyte will have a unique spectrum and thereby a particular PDF associated with it. The skewness, kurtosis, and entropy of the spectrum will tend towards a normal distribution for the slight variations of the PDFs resulting from a single analyte. As the shape of the spectrum changes, so does the HOS and entropy associated with the PDF, thereby resulting in a shift or change away from normal. This work was completed as a top-level analysis of how changing spectra can result in changing distributions of HOS and entropy, thereby producing a measurable parameter for identifying a threshold or change from “background” to “anomaly”.

3.1.1.1 Kolmogorov-Smirnov Test for Normality

The Kolmogorov-Smirnov (KS) test is a nonparametric test to find if the cumulative density function (CDF) of the test data is equal to a hypothesized normal CDF. If the KS test does not reject the null hypothesis, then the CDFs of the two sets of data are considered to originate from CDFs that are not different from each other. In addition to the acceptance or rejection of the null hypothesis, a test statistic (the KS statistic) can also be employed to empirically determine the distance between the test CDF and the hypothesized CDF and can be considered a parameter of how different the two CDFs are from each other.²⁹⁻³⁰

The KS statistic changes as a function of the skewness and kurtosis as this affects the CDF of the test population. To evaluate these changes, distributions with varying skewness and kurtosis composed of 1,000 random numbers determined by the sinh-arcsinh transform were generated. The results, shown in Figure 5, show that there is a minimum of the KS statistic at approximately zero when skewness is 0 and kurtosis is 3, which coincides with a normally distributed PDF (indicated by the red star). The KS statistic is more sensitive to platykurtic (kurtosis < 3) changes and is symmetric about the skewness parameter, as is clearly seen in Figure 5A. Figure 5B displays the symmetrical skewness in a more obvious manner. From this figure, it can also be seen that any one independent change in skewness or kurtosis does not necessarily cause a large change in the KS statistic, changes in both simultaneous effects much larger changes.

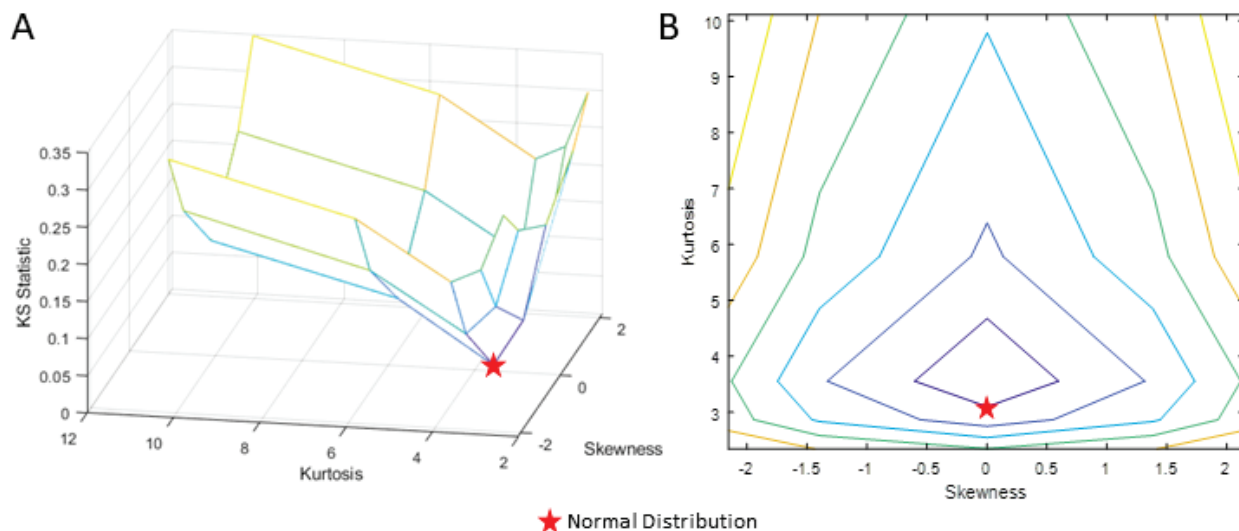


Figure 5: A contour line plot of KS statistic as a function of changing skewness and kurtosis (A) and a two-dimensional representation of A (B). The red star represents the KS statistic of a normal distribution.

The KS statistic changing as a function of skewness and kurtosis is representative of spectral changes as we transform the chemically induced spectrum into a PDF, which the HOS are determined from. This is representative of the spectrum changing by introduction of a new material that effects a large change on the measured chemical spectrum, but is only one way in which the spectrum and thereby the PDF may change. Other estimates by which the PDF may change are outline in the following section.

3.1.1.2 Effects of Bimodal Transformation on HOS and Entropy

A spectrum and therefore the underlying PDF changes through several possible mechanisms. The two main methods by which a change in the data is through a gradual change in an increase in analyte present or via a sudden large change in the amount of an analyte or material that was present. The latter could be induced by transitioning from concrete to asphalt or if there is sudden large change due to analyte being present when it was not previously there. The former could be due to a dissemination on the surface being analyzed increasing in concentration (mass per unit area) as the sensor approaches a detonation site, for example. These two types of changes and how it effects the HOS and the entropy are evaluated more closely here.

Two PDFs progressing away from each other in magnitude of the parameter could result in a seemingly singular unimodal distribution separating into a bimodal distribution and then to two separate unimodal distributions. This, and the way HOS, entropy, and the KS statistic are affected are shown in Figure 6. Two normal distributions with 10,000 random numbers were generated and simulated to gradually separate from each other, as shown in Figure 6A. The first distribution had a constant mean of 0 and a standard deviation of 1 while the second distribution had a mean of 0 to 8 in 1 integer increments and a constant standard deviation of 1. Replicate trials ($n = 100$) of each scenario were replicated with new distributions generated each time.

The HOS, entropy, and KS statistic all clearly change as a function of the second distribution separating from the first. Figure 6B shows the skewness measurement becoming more certain (i.e., less spread in the points) as the two distributions separate, but not deviating from zero. The skewness remaining at zero is expected as the separation of the two distributions of equal number of points is symmetrical. The certainty of the skewness also increases as function of separation and can also be expected. When there is only one distribution present, the skewness is solely dependent on the uncertainty in the generated distribution. However, as they separate, the skewness is no longer dependent on the uncertainty of the distribution, but rather becomes more dependent on the heights of the two distributions, which were set to be equal and therefore not change.

Conversely, kurtosis changes in magnitude as a function of separation, as shown in Figure 6C. The kurtosis decreases, indicating a transition from mesokurtic to platykurtic as the two distributions separate. This is expected as platykurtic distributions appear more “flat topped” compared to a normal distribution. As the two distributions separate, the fit to incorporate both distributions necessarily becomes wider, though the height (i.e., number of points) does not change forcing the fitting parameters to account for wider, but not taller best fits. Additionally, the kurtosis measurement becomes more certain as the two distributions separate, indicating that the certainty transitions from a single distribution to the distance between the mean of the two distributions and the height differences between them.

Entropy decreases slightly as the two distributions separate, as seen in Figure 6D. Entropy is a measure of order or predictability and is maximized when normally distributed,³¹ or stated another way, entropy can be a measure of how uninformative a distribution can be. If entropy is maximized with a normal distribution, as one or more modes become present, the information then becomes more informative, thereby decreasing the entropy. Therefore, our observations of decreasing entropy align with what is expected.

The KS statistic also changes as a function of separation, as seen in Figure 6E. The KS statistic begins at zero when the two distributions are completely overlapped and increases asymptotically to 0.5. The KS statistic is maximized when there is minimal overlap, indicating that their CDFs are maximized in empirical distance from each other. These data were expected as the means of the distributions change, and therefore are less likely to originate from the same distribution.

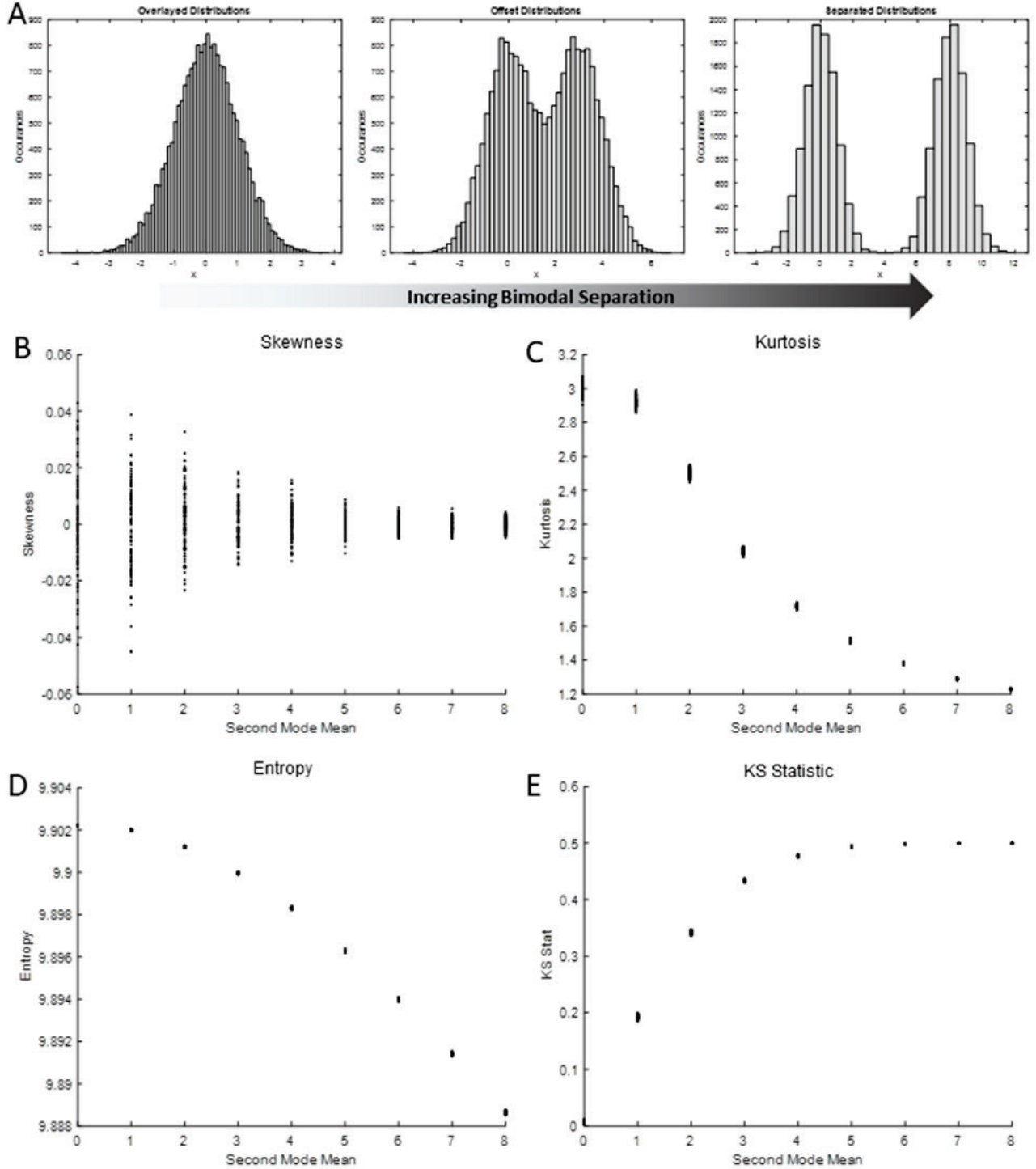


Figure 6: The simulated distributions shifting and separating via an increasing second mode mean (A) and the effects of that separation on the skewness (B), kurtosis (C), entropy (D), and KS statistic (E).

A different way that the distribution may be affected by a change in the spectral response is that a second mode grows in the distribution, and we can relate how this affects the parameters in this work. To evaluate this, a single mode of 10,000 points was evaluated with

second mode growing in 1,000-point increments from 0 to 10,000 at a mean that is shifted by four units and 100 trials with new distributions each time were performed. This is pictorially represented in Figure 7A whereby from left to right a second mode of the distribution is growing until they are equal. The total number of points in both modes of the distribution remained at 10,000 to be representative of what would happen with a sensor that is acquiring the data. That is, the total number of pixels and/or spectral resolution of the detector remains constant during analysis; therefore, the number of elements within any distribution will remain constant.

The skewness changes as expected with the distributions as seen in Figure 7B. Initially, when only one distribution is present the skewness is zero. The distributions generated were normal, so a value of zero is expected. The maximum and minimum skewness values are seen when one of the two modes comprises 10% of the total value and is expected as this is when the two distributions are least alike, causing the fitting parameters to be as skewed to the right (negative skewness values) or left (positive skewness values) as possible at this particular separation. Or put another way, the distributions are least symmetrical at the first increment of 10% (or last increment of 90%) of the second mode. When the number of points in each of the distributions are equal (5,000 points each), the skewness is zero which aligns with the data represented in Figure 6B.

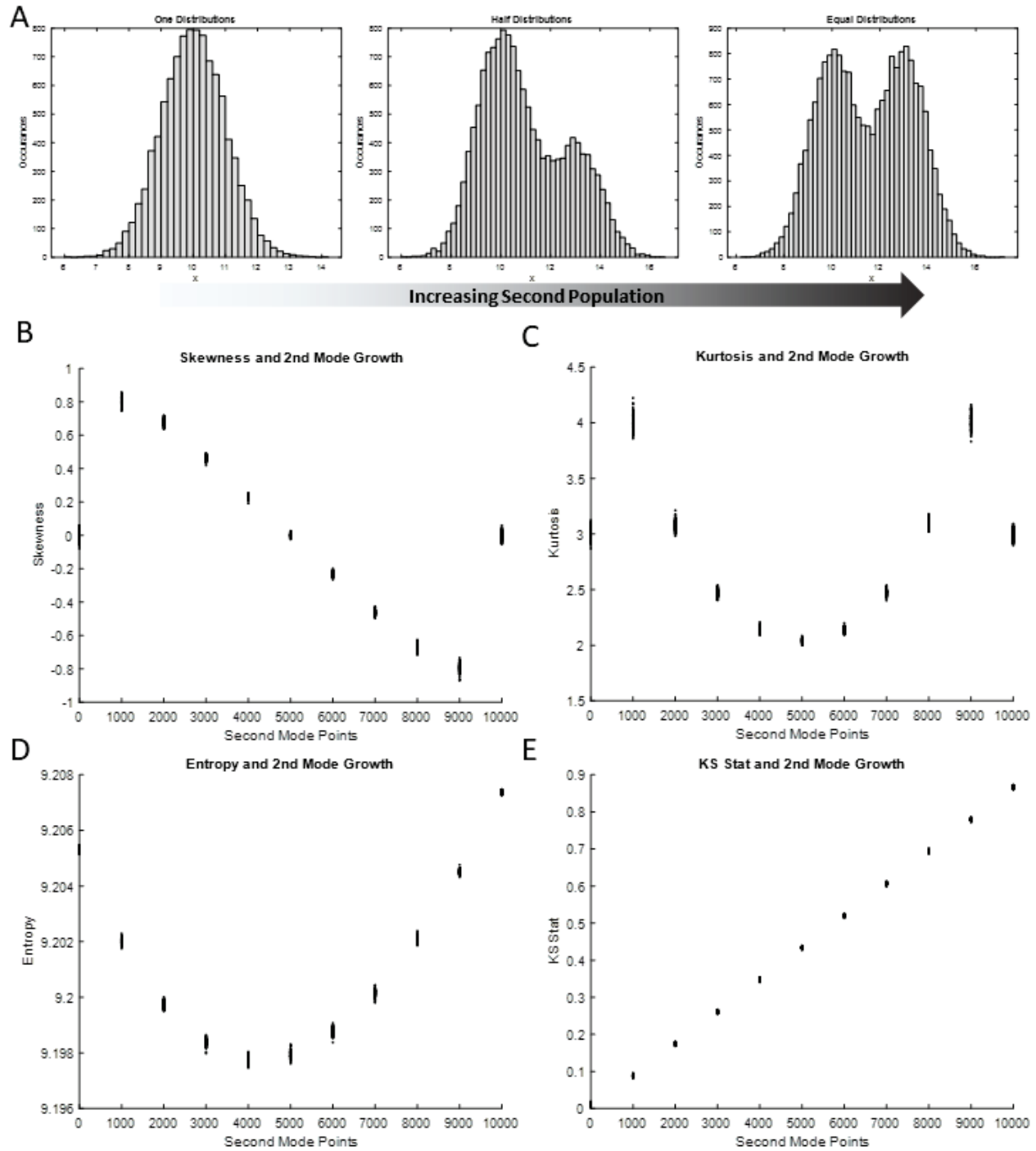


Figure 7: Simulated distributions in which a single distribution has a second mode of the distribution growing while the first mode is reduced (A) and the effects this has on skewness (B), kurtosis (C), entropy (D), and the KS statistic (E).

The kurtosis changes as a function of the second mode growth as well, shown in Figure 7C. The kurtosis begins at a value of three, representative of a normal distribution which is expected. Much like the skewness data, the kurtosis data is symmetrical at 5,000 points, which is when the two distributions are equal. At 10% (the first increment of a second distribution), the

kurtosis reaches a maximum and has the largest variability. The maximum kurtosis is due to the need to fit both the modes of the distribution, and to do so, the fitted parameters require a distribution that would appear taller and narrower, resulting in a kurtosis which is greater than three. When the two modes are equal, the kurtosis is at a minimum due to the fitted parameters requiring a representatively broader distribution to fit appropriately because the two modes are physically separated. The data at 5,000 second mode points in Figure 7C align well with the data represented in Figure 6C where the kurtosis has a value of 2 when the two modes are separated by a mean of 3, as is the case in Figure 7.

Entropy is strongly dependent on the number of elements within a particular distribution. Therefore, maintaining a constant number of elements through the simulation is highly important. In Figure 7D, the entropy can be seen to change as a function of the second mode points. However, it is not symmetrical. The minimum in entropy is seen when the second mode has 4,000 points which was unexpected. Nevertheless, the uncertainty in the spread of the data between 4,000 points and 5,000 points is large, making the two points nearly indistinguishable. As was represented in Figure 6D, there is a smaller entropic value when two distributions of equal magnitude are separated by particular mean value. As entropy is a measure of information available, it was expected that there would be a minimum when the two modes were separated and equal prior to the entropic value change back to what it was originally.

A single sided KS test was performed for these data as well and behaved as expected. In Figure 7E, the KS statistic increases linearly as a function of the second mode growth. The KS test utilized compares the test data to a normal distribution with a mean of zero, which are the same when there is only a single mode represented in the data (0 points in the second mode). As the second mode increases in size and the first mode decreases, the KS statistic is driven higher as it is an empirical distance parameter away from a normal distribution centered at zero.

The kurtosis changes as a function of second mode growth and as a function of separation. Furthermore, it is much less dependent on the number elements in the distribution, unlike entropy which is more highly dependent on the number of points that have been collected. Therefore, we chose to evaluate kurtosis as a method of determining a threshold for transitioning spectrally from one scene to another (i.e., concrete to asphalt).

3.2 Kurtosis As a Threshold

Several methodologies were investigated to determine the best path forward for a threshold to be employed in an anomaly algorithm to mitigate false negative assignments. As data are collected with the instruments we employ, we quickly reach the minimum requirements for the CLT within one or two frames of data (less than one second). Therefore, we utilize CLT to understand what to expect if employing a threshold technique whereby we assume that the parameters collected result in a normal distribution of the values.

3.2.1 Otsu Threshold for Kurtosis

3.2.1.1 Simulated Data

Otsu's method of threshold seeks to minimize intra-class variance when separating a histogram into two discrete classes.³² While originally developed for gray-scale image analysis, this can be applied to any data that may have two discrete classes such as the kurtosis of background changing from one component to another. Therefore, we evaluate Otsu's method for threshold identification in our kurtosis values.

Data was synthesized such that the total number of points were 1,000. The first set was a normal distribution with a mean of 3 and a standard deviation of 1. This results in a spread of data from approximately 0 to 6, as seen in Figure 8A (top) with progressively larger second sets, including the representative set for 3.5% of the data in the second set, seen in Figure 8A (bottom). Representative examples of what the distribution looks like in a histogram is shown in Figure 8B with the red bar representative of the Otsu threshold whereby the intra-class variation is minimized. The representative histograms show the data with second populations of 0, 10, 20, 25, 30, and 35 representing 0.0, 1.0, 2.0, 2.5, 3.0, and 3.5% of the total data in the second population.

The Otsu thresholds for all 100 trials from a second population of 0% to 50% are shown in Figure 8C. Each trial was independent of every other trial such that a new data set was synthesized each time for the populations. Normal distributions with means of 3 and 9 were chosen as the two data sets as they provide clearly separable data when equal in magnitude. The resulting percentage of data required for two clearly separable data sets to become separable with Otsu thresholding is $2.5 \pm 0.2\%$ of data in the second population. This was determined by finding the minimum number of data in the second population in each trial whereby the Otsu threshold was above 5. Therefore, we'd expect if two materials produce two very different spectra resulting in largely different kurtosis values, then when the population reaches 2.5% of the total considered data, this methodology would work in identifying that point as a threshold.

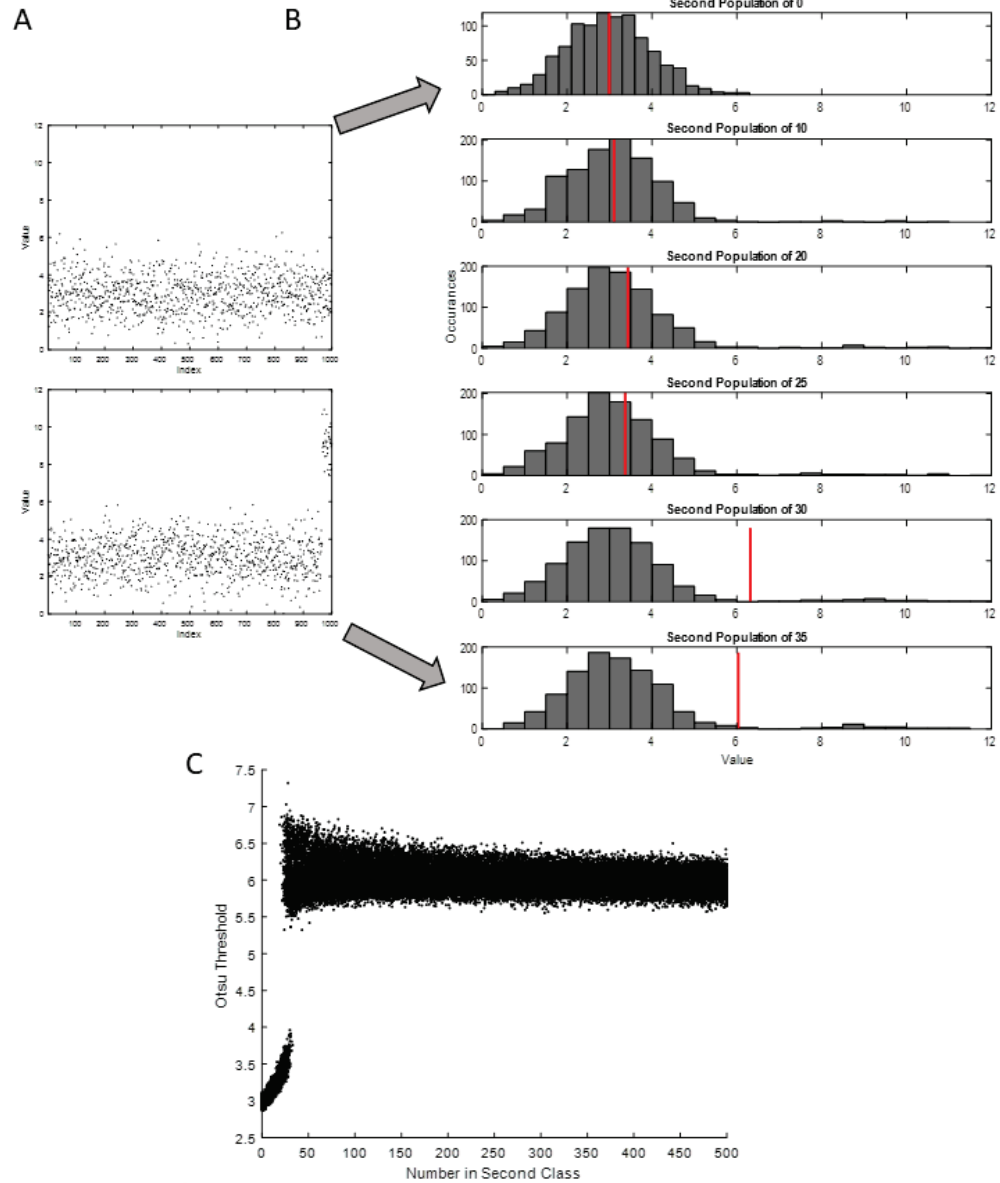


Figure 8: Representative data from two sets with 0% data in the second population (A, top) and 3.5% data in the second population (A, bottom) and selected representative histograms with the red bar representing the Otsu threshold (B). The Otsu threshold for 100 trials for up to 50% of data in the second population (C).

3.2.1.2 Laboratory Data

Simulated distributions worked under the assumption that laboratory or field data would present itself in a normal-type distribution. We therefore evaluated two similar types of materials: aged asphalt and asphalt road, both of which present themselves in normal distributions. These samples were both collected under the same conditions and are representative of two different types of asphalt that may be encountered with analysis. After the data were collected, regions of interest were identified that had no abnormalities (e.g., cracks, discolorations, etc.). Representative spectra ($n = 10,000$) were obtained from the data and the kurtosis values were determined, as shown in Figure 9A. Of the 10,000 spectra, spectral mixing

of 10,00 spectra to make up a simulated expected spectrum of the two was completed. That is, if a spectrally mixed synthetic spectrum of 10% asphalt road with 90% aged asphalt was completed, 100 spectra were randomly chosen from the 10,000 available spectra of asphalt road and 900 spectra were randomly chosen from the 10,000 available aged asphalt spectra, summed, and normalized to create a spectrally mixed spectrum.

After the new spectrum was created, the Otsu threshold was determined for each mixed spectrum. The data are represented in Figure 9B where the horizontal solid blue line represents the average kurtosis value for the aged asphalt and the dashed blue lines represent ± 1 standard deviation. The solid and dashed red lines represent the mean and standard deviation of asphalt road. The resulting Otsu threshold can be clearly seen migrating from aged asphalt when the representative percentage of the spectra are aged asphalt, clearly separating the two through the majority of the graph and then settling on the mean kurtosis for asphalt road when the spectra are primarily that of asphalt road. This aligns with what was presented in Figure 8 as well though the simulation only progressed halfway. The sensitivity to the change in the histogram is not as high as in Figure 9B as it was in Figure 8C, however, the data still show the same trend. One reason why this may be is that the two populations were not as well-separated, resulting in a larger amount of the second population needing to be present prior to the threshold fully minimizing the intra-class variance.

The histograms in Figure 9C show the two populations or classes that the Otsu method is trying to minimize the intra-class variance for. The top left of Figure 9C shows pitfalls of using the Otsu method; the minimization of the intra-class variance is split through the mean of the kurtosis values. It's not that there are two classes or populations that we are trying to separate here, but rather the method assumes that there are two classes or populations to split even if that is not true. Therefore, other precautions would need to be taken to ensure that we aren't unnecessarily dividing the data. However, through the other histograms of Figure 9C, the Otsu threshold method appears to work very well until 100 percent of the other spectrum is substituted (bottom right), whereby we have the same issue as when 0% of the spectrum is substituted. As the data become more overlapped, it will become increasingly difficult to separate via Otsu's method of thresholding.

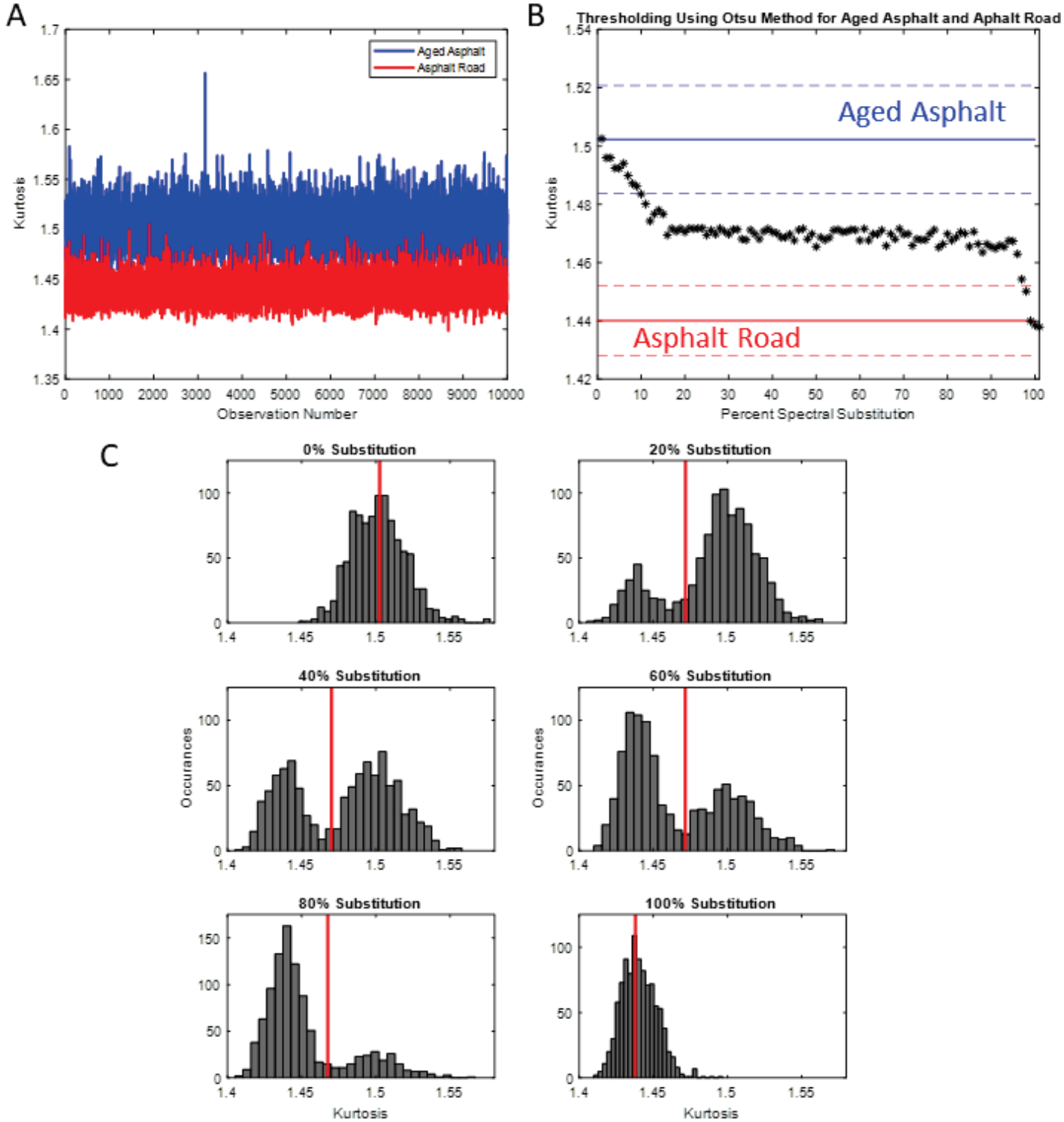


Figure 9: Laboratory collected data for kurtosis of aged asphalt (A, blue) and asphalt road (A, red) from 10,000 representative spectra. The Otsu threshold (*) from spectrally mixing the two spectra with the average kurtosis $\pm$ the standard deviation represented by the solid and dashed lines, respectively for each set (B). The representative histograms of the Otsu threshold with the kurtosis data (C).

3.3 Parameterization of Spectral Data

The instrument that was used in this work has dual detectors, one of which is optimized for VNIR and the other for SWIR light. An initial data set in the SWIR that was used to determine the utility in HOS and entropy for separating materials was one of white powders which included ammonia nitrate. Figure 10A is the representative image obtained from the hyperspectral line scanner with AN, WC, GA, T, S, SC, SBC, and their representative regions of interest shown as the red boxes. Each material was placed in a Petri dish so that the powders were contained and well defined. The spectral information can be represented as the skewness, kurtosis, and entropy. These parameterized spectra can be represented on a three-dimensional scatter plot of all the data to show how they are separable. It can be seen in Figure 10B that the data are largely visually separable without the use of advanced statistics. In particular, the SC, WC, and SBC are visually apart from the other materials whereas the ammonia nitrate overlaps some. By employing additional statistical and machine learning (ML) techniques such as parallel coordinates (this section) or DBSCAN (see Section 3.4.1.1.1), the data can more easily become separable, and we can begin to build an algorithm to better separate and analyze the data.

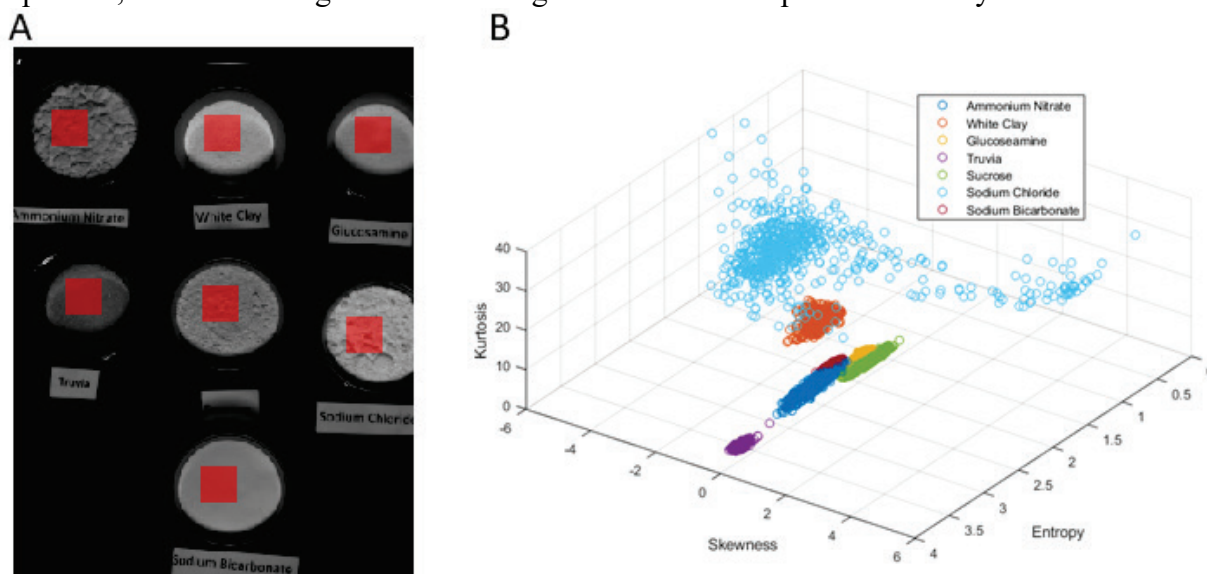


Figure 10: A representative image of the regions of interest identified for each analyte (A) and the parameterizing of the spectra within those regions for skewness, kurtosis and entropy shown in a three-dimensional scatter plot (B).

Identifying the capability to separate through parameterization is one aspect of understanding which sensor provides the most separable data and most likely method for detecting an analyte on a background, i.e., anomaly. As the focus of this effort is on explosives and anomaly detection, the various analyte and background data were acquired and visualized to better understand the sensor type and which materials separate the most with the data provided. The analytes of interest for explosives were AN and PC. To distinguish the potential background materials and to make bounding the problem easier, we separated “background” materials into two categories of “ground” and “surface” to represent background materials that may be a surface material (e.g., RCP, BCP, WCP, RM) or a ground material (e.g., G, RD, AS, BC, SMC). These groupings were visualized with parallel coordinates to better understand which sensor and which parameter would be best to differentiate the analytes of interest from everything else.

Parallel coordinates is a method for visualizing high dimensionality data sets to better understand which dimension provides the best separability.³³⁻³⁴ Generally speaking, in parallel coordinates, the more separate an individual line is from the rest of the lines, the easier it is to separate with that dimension or parameter of the data. In Figure 11A, the parallel coordinates of the ground materials and analyte are shown in six dimensions or parameters, which are the entropy, skewness, and kurtosis in the VNIR and SWIR region. Here, AN is more differentiable from all of the other materials in the SWIR with HOS and entropy. This is demarcated by the blue line with the blue circles showing the coordinate value clearly above or below the other data. In the zoomed in inset (Figure 11A, right) PC (orange line, green circles) is less separable, and largely overlaps with asphalt (purple line). Figure 11B highlights the analytes with different surfaces and shows that AN is again more separable in the SWIR but largely with entropy and skewness. PC marginally overlaps with RM in entropy but is separable from the other materials with SWIR entropy. Overall, it appears as if the SWIR provides the most information for separability with these analyte/material combinations. This aligns well with other data we've seen empirically, including previous unpublished data showing separability with HOS and entropy from 550–1700nm. These data are suggestive that VNIR information does not provide as much separability and therefore may be able to be eliminated as a spectral region of interest in the future.

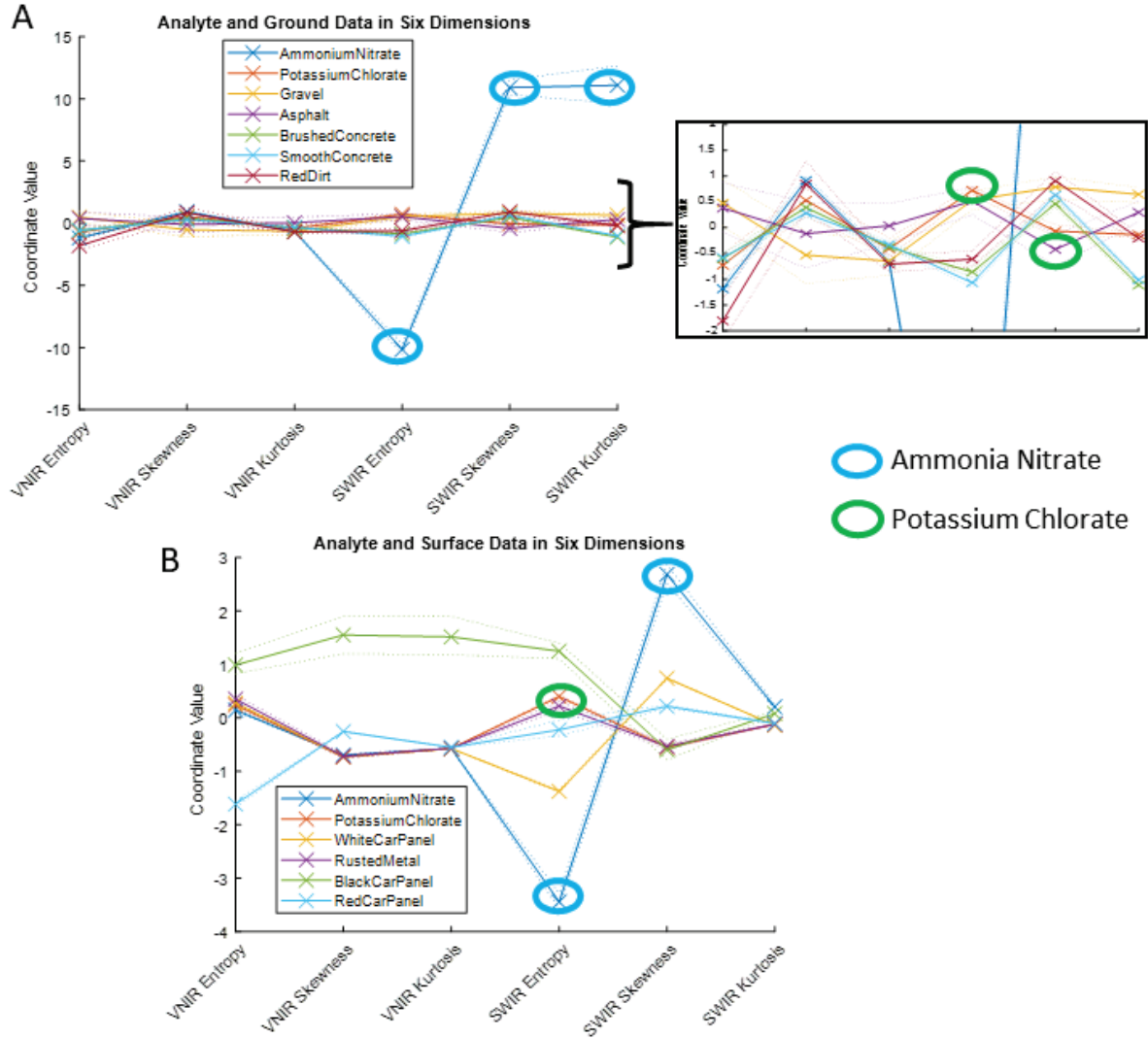


Figure 11: Parallel coordinates of ground materials in six dimensions and a zoomed in area to show differentiability of ammonia nitrate (blue circles) and potassium chlorate (green circles) (A). Parallel coordinates of surface materials highlighting ammonia nitrate and potassium chlorate (B).

Through understanding the parameterization of the collected data, we can see that the SWIR is more robust in differentiating HOS and entropy than the VNIR. This is suggestive when an algorithm is developed and employs the data streams from the instrument described herein, that the SWIR region will be more robust at detecting anomalies than the VNIR. Additionally, these results also suggest that instrumentation that employ SWIR or a combination of VNIR and SWIR (not just VNIR), will also perform reasonably well in detecting anomalies if HOS and entropy are employed as the parameters. Delving deeper into these ideas and how to characterize the data are the subjects of the remainder of this work.

3.4 Real Time Algorithm Development

The goal of this program was to develop an anomaly algorithm that could perform in real time on hyperspectral data. The challenges, as outlined in the introduction, are that there

is an enormous amount of information that is produced with hyperspectral data. While there have been many efforts in data reduction to increase speed of processing, many methods simply utilize a group of peaks and still look for the match of those peaks and often require corrections and pre-processing.^{13, 35} We step away from this by utilizing the PDF, as discussed above, comparing only components of the PDF or parameters derived from it. The PDF still contains the information from the spectrum that was originally captured, but in a compressed form. The details of the algorithm and the ability to process in real time are provided in the following sections.

3.4.1 Anomaly Algorithm

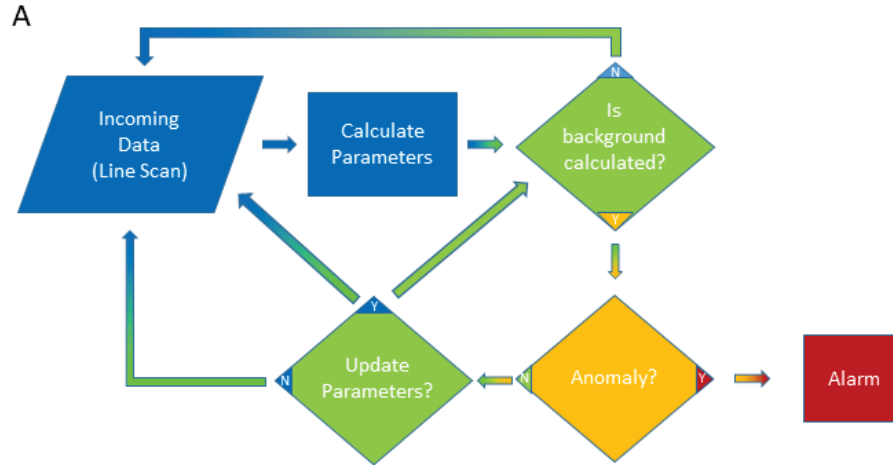
The algorithm was specifically designed to identify class differences and not chemical specific differences. This allows us to capitalize on utilizing the PDF and not the spectral peaks. The algorithm utilizes semi-supervised ML techniques and can therefore be broken into two sections: a training section and a testing section. That is, the algorithm is trained on initial data collection and is labeled as background or “non-anomaly”. The second set of data is test data and has no labels associated with it. Figure 12A shows the block diagram for the algorithm that was employed. First, the incoming data in the form of a line scan was obtained which consisted of 640 pixels of 273 wavelength data points in each pixel in the VNIR and 646 pixels of 269 wavelength data points in each pixel in the SWIR. This incoming data was reduced from the 273 and 269 wavelength data points per pixel to 3 parameters per pixel, a 98.9% decrease in information usage by determining the skewness, kurtosis, and entropy from the PDF associated with each pixel.

3.4.1.1 Background Determination

The background is determined by the training set of data. This set of data is predetermined in length and must be sufficient to appropriately cluster the background parameters. While one line scan may be sufficient, the typical number of line scans for background acquisition is between 10 and 50. In Figure 12A, the first green diamond in the flow chart is the point in which it is determined if the background is calculated. If it is not, the algorithm simply goes back to Step 1 (Figure 12B). Background collection continues until the condition for the number of line scans are met. After an appropriate number of background lines are collected, the training data is analyzed (Figure 12B, Training Data Step 3).

3.4.1.1.1 Analysis of Training Data

To analyze the training data, a density-based spatial clustering of applications with noise (DBSCAN) algorithm is employed. A full description of how the algorithm works is outside the scope of this report, but a good review can be found by Schubert et al.³⁶ The DBSCAN algorithm uses a minimum distance parameter to determine closeness of adjacent points, and if they are within that minimum distance are clustered together. There are also other cases in which data points can be connected but not part of the cluster and labeled as just noise. One of the advantages of DBSCAN and algorithms similar to it is the data points can be labeled as “noise” whereas other clustering algorithms necessary assign a cluster (defined by distances from points or centroids) to each point. After the clusters are identified in training data, a convex hull for each cluster is determined to rapidly determine if a subsequent test data point is within or outside of the convex hull.



B

Algorithm

Training Data (Background)

- Step 1: Load and energy normalize it
- Step 2: Calculate parameters (HOS and entropy)
- Step 3: If number of background lines condition met, determine clustering for centroid locations; else continue to collect background

Test Data

- Step 1: Load and energy normalize it
- Step 2: Calculate parameters (HOS and entropy)
- Step 3: Determine if test data falls within cluster of background
- Step 4: Label anomalous points to alarm
- Step 5: If parameters need to be updated, update them; else return to Step 1

Figure 12: A block diagram of the algorithm used within this work (A) and pseudo code of the algorithm used within this work (B).

There are several parameters within the DBSCAN algorithm that can be optimized for performance. These efforts will be discussed further in subsequent sections. In short, the epsilon factor, minimum elements in cluster, and minimum distance between points can be changed by the user and therefore optimized in performance. The number of points that make up the training data can also be optimized as well. Generally, as the number of points in the training data increases, the longer it takes to determine all of the cluster locations and therefore, there is a lag between going from the training data to the test data.

3.4.1.1.2 Determination of An Anomaly

Once the training data is collected and the background clusters are determined, each subsequent group of parameters from the line scanner is tested against the clusters. Since the convex hull is calculated in three dimensions, a determination of if a data point falls within one of the potentially many convex hulls is a rapid and straight forward calculation. If the data falls outside of the convex hull, then it is considered an anomaly.

From this point, several other actions could be taken such as pass the location information of an anomaly to a secondary processor for further and more computationally burdensome calculations or background parameters could be updated. However, we have not fully implemented either one of these. The thresholding discussed previously is one possible method for wanting or needing to update the background parameters. That is, if the parameters fall outside of what could be considered normal and CLT could be safely assumed, then an update to the background parameters may be merited.

3.4.1.2 Real Time Application

Demonstration of real time capability was first completed through a real time simulator (RTS). The RTS that we used tested our capability of analyzing data in real time with real data while employing our algorithm. Upon validating through simulation that data could be processed at faster than 0.1s per frame, our algorithm was used with the instrument to record the streamed line scans and analyze them in real time.

3.4.1.2.1 Real Time Simulator

The RTS was designed such that previously collected data could be analyzed and tested for real time analysis capability. Two instances of MATLAB were necessary when using the RTS, as shown in Figure 13. The left side of Figure 13 describes the first instance of MATLAB whereby the data stream is created. This is first done by loading the hypercube from a previous data collection event. Loading the hypercube is prohibitive to employ in line with the streaming process because the file sizes are so large. After the hypercube is loaded, a memory mapped file is created in MATLAB. A memory mapped file is a virtual file location that provides the ability to both write and read simultaneously. Utilizing a memory mapped file mitigates any operating system restrictions and allows for us to freely write and read from the file. The memory mapped file is written at a set rate, equivalent to a frame rate from the sensor. This file acts as the live data stream and continues until the last line of the previously loaded hypercube is written into memory.

Once the memory mapped file is created, a second instance of MATLAB can establish a connection with the memory mapped file and begin identifying if a new line has been written to the file, as seen in Figure 13 on the right side. The two columns of Figure 13 are offset to represent the timings associated with the program; step one in the second instance of MATLAB cannot start until step two of the first instance in MATLAB is completed. After a new line is identified, our algorithm is employed just as it would be through obtaining live streamed real time data from the instrument and the processes is completed until the end of the file is reached and the process times out.

MATLAB Instance 1 (Data Stream)	MATLAB Instance 2 (Process Stream)
1. Load previously obtained hypercube	
2. Create memory mapped file	1. Access memory mapped file
3. Write a single line of hypercube to memory mapped file	2. Search for new line written
4. Pause for selected amount of time	3. Process line of hypercube
5. Repeat steps 3-4 until end of file is reached	4. Repeat steps 2-3 until end of file is reached and process times out

Figure 13: Pseudo code for RTS with two instances of MATLAB. The first instance (left) loads a hypercube and creates a data stream to be processed (right).

The results show that we are theoretically capable of analyzing data that is being streamed at a rate of 10 Hz. Figure 14A shows the memory writing speed (left) and the signal speed (right). The memory writing speed is the time it takes to write a single line from the hypercube to the memory mapped file. The data show on the left side of Figure 14A that the processing time to write each line is less than 0.035s, with the exception of the first line, which was reproducible and less than 0.05s. The initial point appears to take longer than the rest of the points as the memory is initially written into the space. However, the timing is still less than that of the 0.1s threshold target we aimed to achieve.

Since the data can be written quicker than the 0.1s target threshold, a pause was put into place within the first instance of MATLAB, after step 3. The pause simulates the frame rate at which the instrument will be streaming the data in order to appropriately understand our capability in processing at real time speed. The right side of Figure 14A is the signal stream speed. The data show that each line is written in less than 1.5ms with the majority of the lines written in less than 0.5ms. This timing was calculated from the end of the previous line that was written to the file to end of the current line written to the file. Therefore, the simulated frame rate is consistently within 1% of our target threshold to simulate the frame rate of the instrument we are testing. Figure 14A shows that we are capable of streaming simulated data at the frame rate required for the program.

Figure 14B shows the timings from processing the data in the RTS. The computation time (black solid lines) are broken into training data (Figure 14B left) and the test data (Figure 14B right) as the processing is different for the two. There were 650 lines used in background/training data, as shown in Figure 14B (left) with a computation time increasing as the number of lines increases (slight positive slope). This is to be expected as the data are aggregated together as the training data are acquired. After line 650, the clustering portion of the

algorithm initiates and we see the processing time increase. The increased processing time due to the clustering portion of the algorithm is shown in Figure 14B (right) and is just over 0.1s. However, all of the lines are processed at less than 25ms per line for the remainder of the data, which is about a quarter of the threshold of 0.1s (red dashed line). Also of note, is that after the algorithm is trained, the time it takes to process each line is constant. This suggests that the RTS could continue indefinitely after it is trained and will provide real time information on the processed lines. These data confirm our capability of programmatically remaining at less than 0.1s per processed line for our algorithm.

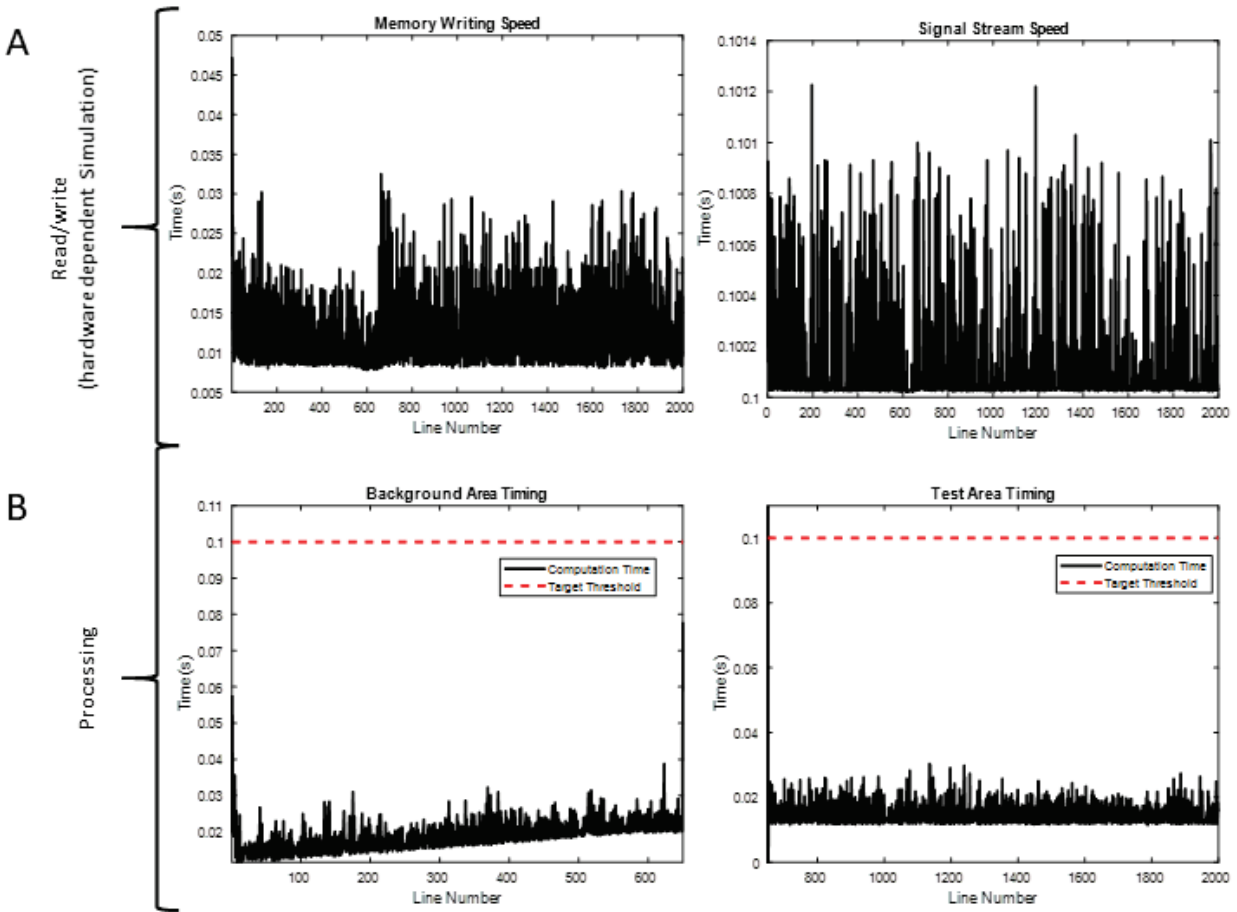


Figure 14: The data stream produced by the RTS showing the memory writing speed and the signal stream speed (A) and the timings associated with processing the training data/background (B, left) and the test area (B, right) with a red-dashed line representing the 0.1s threshold goal we had.

3.4.1.2.2 Real Time Hyperspectral Line Scan Analysis

To implement the algorithm and real time processing in line with the instrumentation, significant work had to be done to modify our existing instrument by the manufacturer to stream the data in real time via a Camera Link connection. Once this work was completed, the software development kit (SDK) provided by the manufacture was used to implement the necessary programmatic calls to begin streaming the data in real time within MATLAB. The SDK was not as flexible as we were anticipating upon preliminary evaluation.

One potential hurdle that was recognized is that data can't "stream" from the instrument and has to be called. Therefore, a programmatic frame rate must be instated and this frame rate remains the same for both sensors since they are in the same loop. Despite this limitation, real time processing was realized.

The core of the program implemented was the same algorithm as the RTS. Figure 15 shows the results from real time acquisition with the instrumentation in a laboratory setting, displaying the timings associated with 150 line scans (black lines) and the associated threshold (red dashed line). There are large increases in the processing time when DBSCAN is initiated within our algorithm and a small increase in the timings at the same time when we sent a signal to the stage that the instrument was on. However, we are not sure if that is the root cause of the increase at the 100th line scan or not, as Figure 15B also has periodic instantaneous increases in processing time.

The amount of processing time is higher than we had anticipated based on our RTS, with the timings averaging about 0.12 s as seen in Figure 15A. These results, while drastically different from what the RTS provided is only about a factor of four higher. This is because the RTS was developed and tested to process one row of data from one sensor. The sensor employed in the lab has two sensors running simultaneously. Each sensor in the instrument is sending a data stream that is processed. Therefore, the estimated process time for a single data stream is approximately 0.06s, within our target threshold. An additional future study will need be completed to confirm this.

A long-term study of data processing is shown in Figure 15B. Data were processed in the same way as in Figure 15A but for approximately 16 longer. In addition to 2500 line scans, resulting in just under 5 minutes of data stream being processed, the data also show an approximate amount of data processed. Over less than 5 minutes, 7GB of data is processed without a measurable increase in processing time after the initial clustering. The origin of the periodic and discrete jumps in processing time shown in Figure 15B is unknown, but may originate from reallocation of random access memory (RAM) or the processor queuing some event. While this does not directly impact the algorithm's performance, this results in one or more missed incoming line scans.

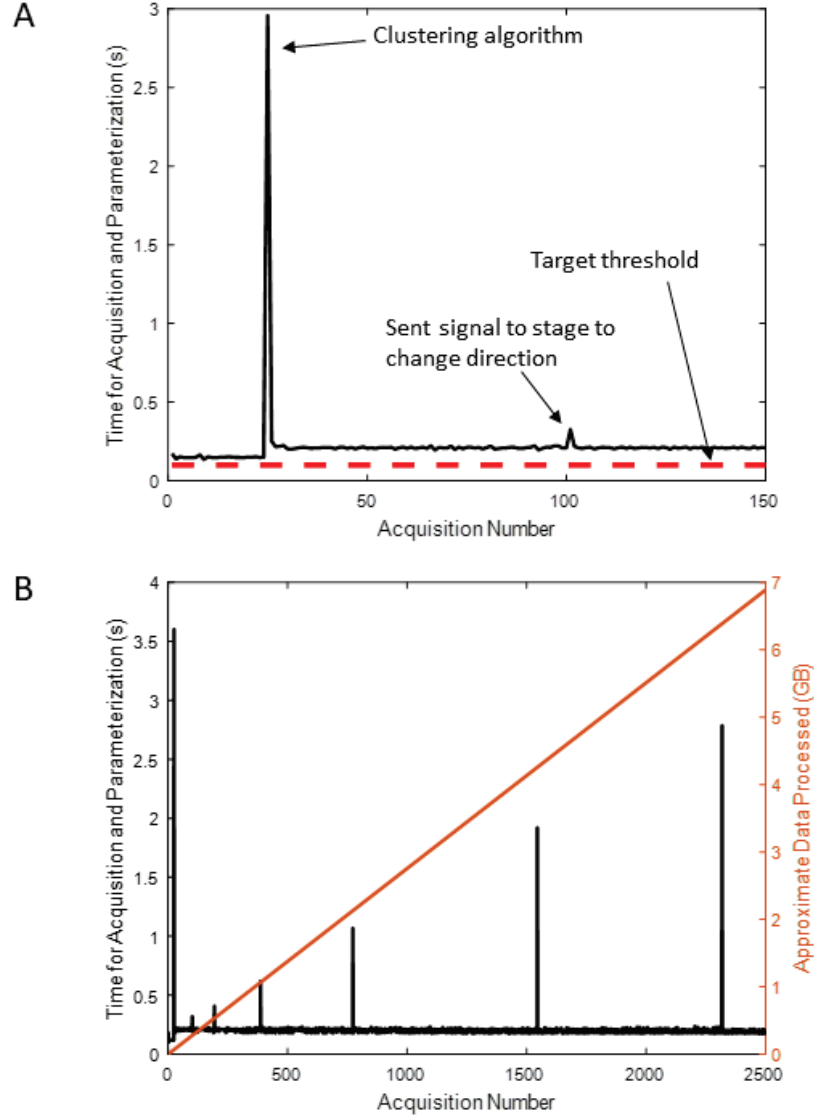


Figure 15: Real time operation of the instrument with the anomaly detection algorithm employed for 150 line scans with respect to the target timing (red dashed line) (A), and a long-term collection of 2500 lines and the coinciding approximate amount of data processed within that time (B).

The anomaly detection results are presented in Figure 16 from the data presented in Figure 15A. Figure 16A is a labeled representation of the analyzed area whereby the gray areas are concrete coupons laid out in a grid, the red areas are a red car panel placed within the concrete coupons, the black area is a black plastic mat that the materials were on, and the white area is a white car panel. The algorithm was trained on the concrete coupons in the first 25 line scans and as a result, we expected that the areas that have the concrete coupons will be identified as background or not anomalous whereas the other areas are expected to be anomalous. This is shown and validated in Figure 16B whereby only the concrete coupons are identified as not anomalous, and all of the other materials are considered an anomaly with the SWIR sensor in the instrument.

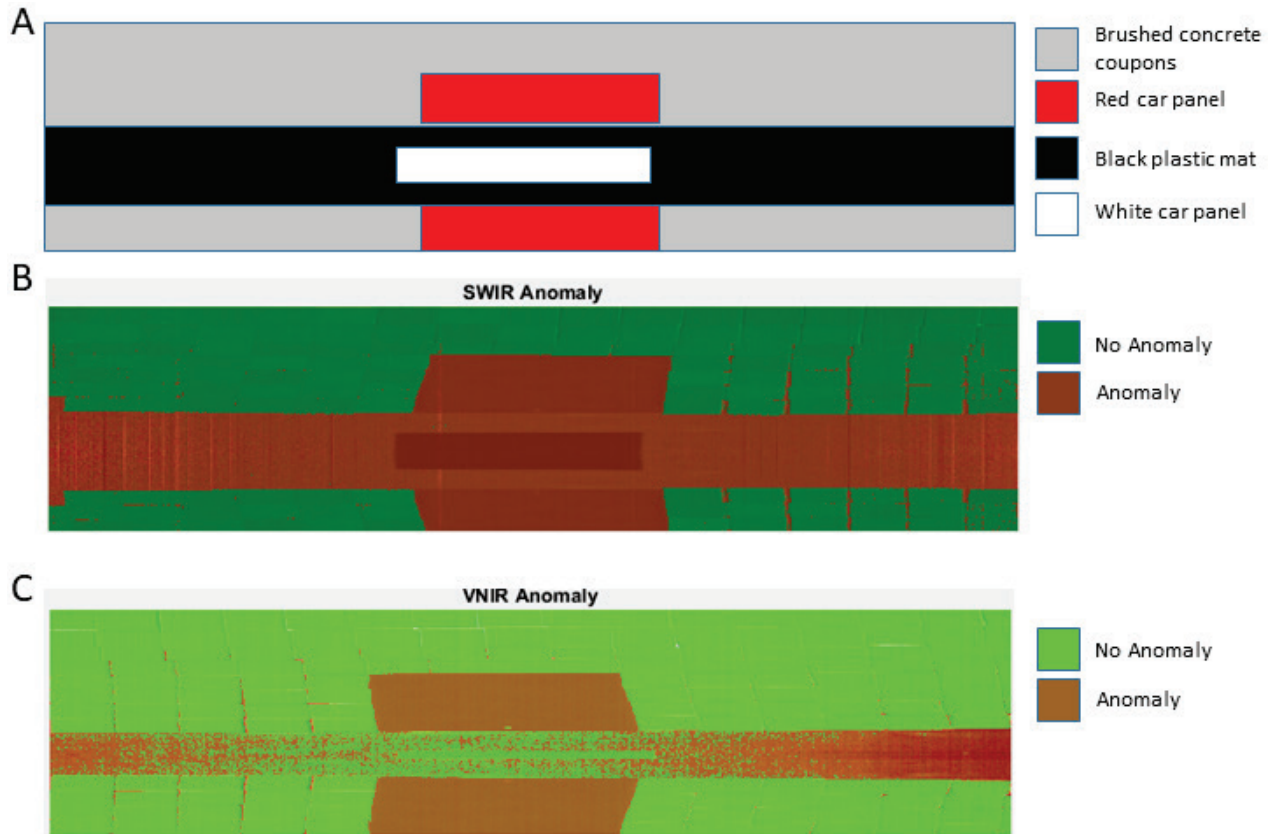


Figure 16: Labeled representation of real time anomaly detection results (A), the SWIR anomaly detections (red) and areas considered not an anomaly (green) (B), and VNIR anomaly detection (red) and areas considered not an anomaly (green) (C).

The VNIR data show the RCP identified fully as an anomaly, but fails to fully identify the black plastic mat or the WCP, as shown in Figure 16C. This is generally in line with that was shown and discussed with Figure 11. The VNIR data does not generally differentiate materials as well and therefore makes detection of anomalies more difficult as multiple materials appear more similar to each other. These data presented in Figure 15 and Figure 16 show the prospect and usefulness of the anomaly detection algorithm we employ in being operable over longer periods of time and processing in real time.

3.5 Ground Truth Analysis

While it can appear that the anomaly detection algorithm is performing well from a visual standpoint, the only way to quantify and validate how well the anomaly detection algorithm is performing is by providing a ground truth data set. In a ground truth data set, all of the locations of anomalies are known in advance and can be correlated to the decision the algorithm makes about each individual point or pixel. Therefore, we developed a ground truth data set with different “concentrations” of analyte for simulating spectral content and therefore the performance of the algorithm.

The synthetic ground truth image has known areas of anomaly which are defined by the mass-density that is alphanumerically represented as shown in Figure 17A. For example,

the first area is a solid square that is 44x44 pixels in line with an alphanumeric descriptor of “1g/m²” indicating that the anomaly present in that area is of a mass density of 1g/m². All of the black areas in the binarized mask in Figure 17 represent where anomalies are located and are designated by the mass density located in the right column of the ground truth synthetic image. The last line “US ARMY DEVCOM CBC” is representative of only analyte and no “background”. Some of the key characteristics of the synthetic ground truth image are shown in Figure 17B. These are a solid 44x44 pixel square to represent a “large” area of contamination, a 5x3 pixel caret to represent a small singularly connected group of pixels, and a 6x8 pixel area that has a whole in the middle and only one pixel width on the top and the bottom. Synthetic mixed spectra are applied to each area as designated by the alphanumeric representation and is described in the next section. This methodology can be applied to any binarized image.

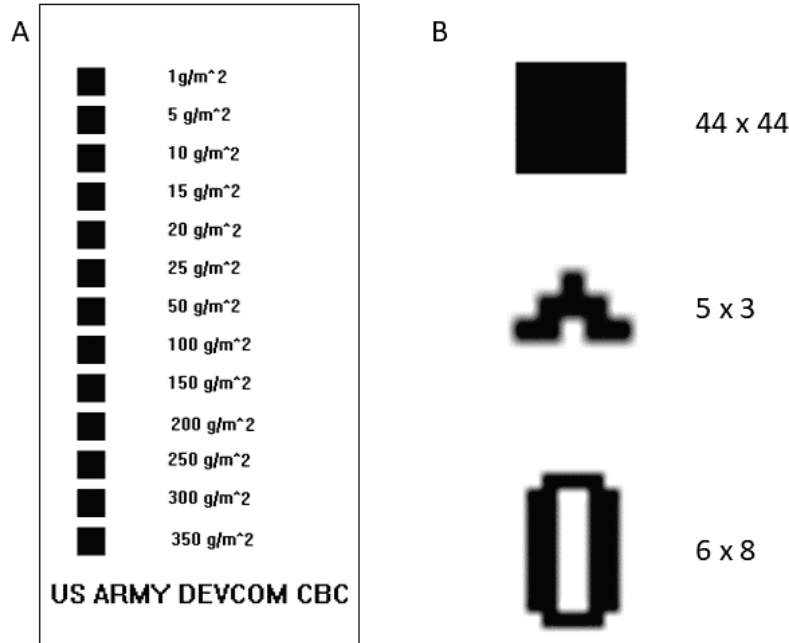


Figure 17: A binarized mask of the developed ground truth data set with both alphanumeric descriptions and square areas of where an anomaly is present (A) and a magnified view of various aspects of the ground truth (B).

3.5.1 Synthetic Spectra for the Ground Truth

Spectra were simulated based on a binary summation of spectra that compose each area of at a specific mass density. To appropriately simulate spectral composition, simulated disseminations were completed from 1g/m² to 450g/m². The details of these simulations are described elsewhere by the authors.³⁷ Figure 18A shows the resulting dissemination patterns, which appear as expected. The density of the droplet size increases as the mass in the dissemination increases from left to right. The area of droplets with respect to the area analyzed was calculated and a linear relationship between the two was realized, as shown in Figure 18B. This relationship can be used to simulate spectra if we make several assumptions. The first assumption is that the droplet material and the background are spectrally opaque materials. The second assumption is that they have similar optical cross sections. The third assumption is that overlapping droplets do not increase the spectral content. The fourth assumption is that there is uniform coverage across the region of interest that is interrogated by the sensor.

The simulated spectra used in the ground truth data set were composed of a binary summation of spectra based on the required percentage of each to meet the specific mass density target as described in Figure 18B. In all the simulated spectra, experimentally obtained spectra were used to create a new, simulated spectrum. That is, each pixel in the ground truth was composed of 1,000 spectra picked at random from a pool of experimentally captured spectra. The pool of spectra were collected from the background and target analyte in advance. The linear relationship described above dictates the number of spectra chosen at random from the pool of available spectra. Therefore, a pixel that required 23% analyte and 77% background to meet a specific mass density would result in 230 spectra chosen from a pool of spectra via a random permeation and 770 spectra from the pool of spectra representing the background were chosen via a random permeation. The 1,000 spectra were then summed and normalized resulting in a simulated spectrum used in the ground truth data set.

This particular method could be applicable to any combination of analytes and backgrounds so long as at least one spectrum of the pure material was obtained. Having multiple spectra to choose from will incorporate more of the variability in the measurements and may be more realistic, but that may not always be practical.

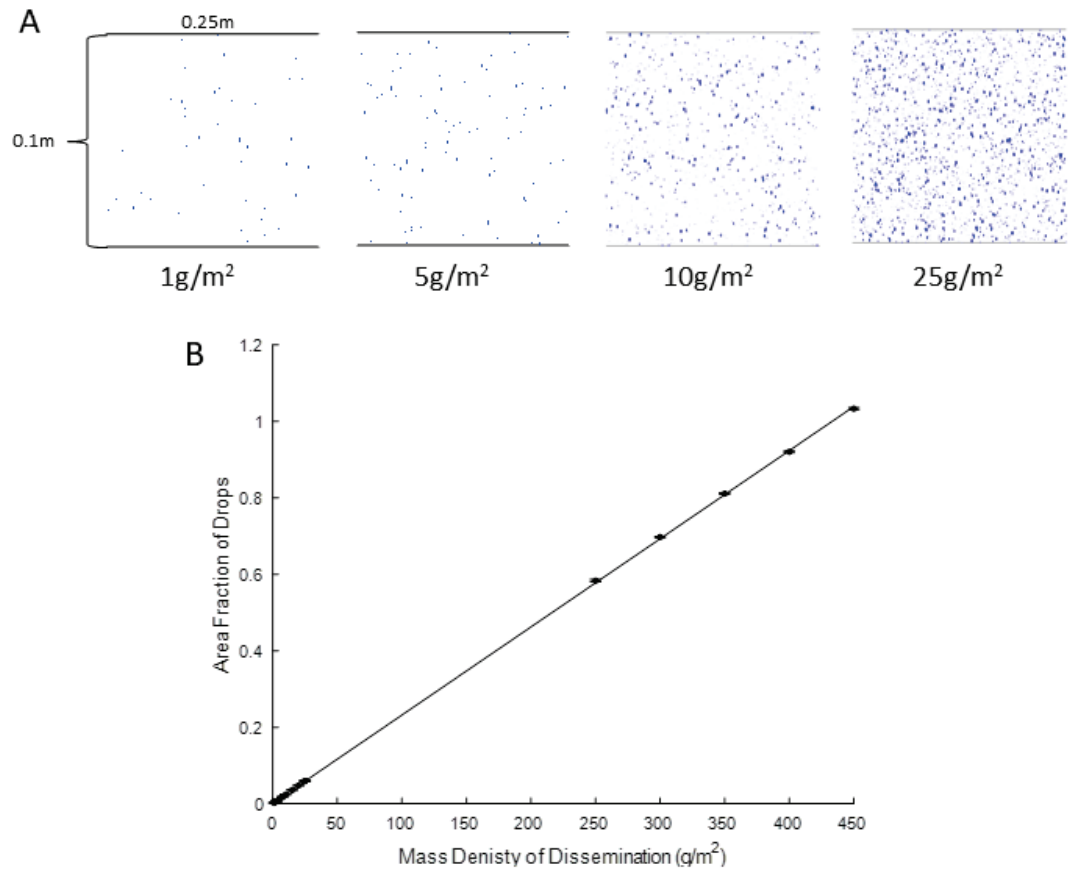


Figure 18: Simulated dissemination patterns of 1, 5, 10, and 25g/m² (A) and the correlated area fraction of drops as a function of mass density (B).

3.5.2 Ground Truth Performance

Each pixel within the ground truth had a simulated spectrum determined for it. Once this is completed, the resulting data represents a three-dimensional hypercube of information, analogous to what would be obtained with consecutive line scans on the hyperspectral line scanner. This data set can then be analyzed with our algorithm and specific metrics can be extracted to understand performance.

Metrics associated with receiver operator characteristic (ROC) curves can be employed for comparative analysis. That is, the label of each pixel is known and therefore we can determine how well the algorithm performs. We can use relationships for the designation of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) to determine recall, specificity, false positive rate (FPR), and limit of anomaly detection (LOAD). These are defined as follows:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$Specificity = \frac{TN}{TN + FP} \quad (4)$$

$$False\ Positive\ Rate = 1 - Specificity \quad (5)$$

$$Limit\ of\ Anomaly\ Detection = FPR + (3 * \sigma_{FPR}) \quad (6)$$

Recall, specificity, the FPRs are well known parameters used within ROC analysis.³⁸ The LOAD is based on using a ground truth mask such as the one proposed above. Each section or concentration area of the mask can be independently defined, and the false positive rate of each area is defined by the noise associated with our anomaly detection algorithm. That is, the FPR are the number of pixels that are designated as an anomaly but are not. Increasing the FPR too greatly results in lower differentiability between actual anomalies (TP) and not (FP). Limit of detection, an analytical figure of merit, is traditionally measured as the signal of the blank summed with three times the noise level of the blank, and is said to be the limit of detectable signal.³⁹ Therefore, this is directly applied to our anomaly detection capabilities with the FPR acting as the noise resulting in the LOAD as described above.

This logic can be applied to the ground truth and analyzed for the limit of detection of a particular analyte/background combination. As shown in Figure 19A, a binarized mask has concentrations assigned to the black areas based on the alphanumeric representation. Figure 19B is a scaled version to visually indicate the concentration gradient of the ground truth data. Figure 19C shows the results from our anomaly detection algorithm using simulated ground

truth for AN as the analyte and AS as the background in the VNIR. It is clear that the LOAD of this particular combination is 5g/m^2 , which is visually evident at the black square on the left side of where the alphanumeric representation is located.

The recall (Eq. 3) is often used to determine the goodness of a result. This could be used as an alternative to identifying the LOAD as maximizing recall suggests a minimization of false negatives. Which, for CBRNE applications maximization of recall is important as maximization of this value results in all of the potential anomalies being characterized as an anomaly. Through our purposeful creation of the mask for a ground truth, we realized theoretical LOADs as shown in Figure 19. This method can be modified for any concentration range needed and for any analyte/background combination.

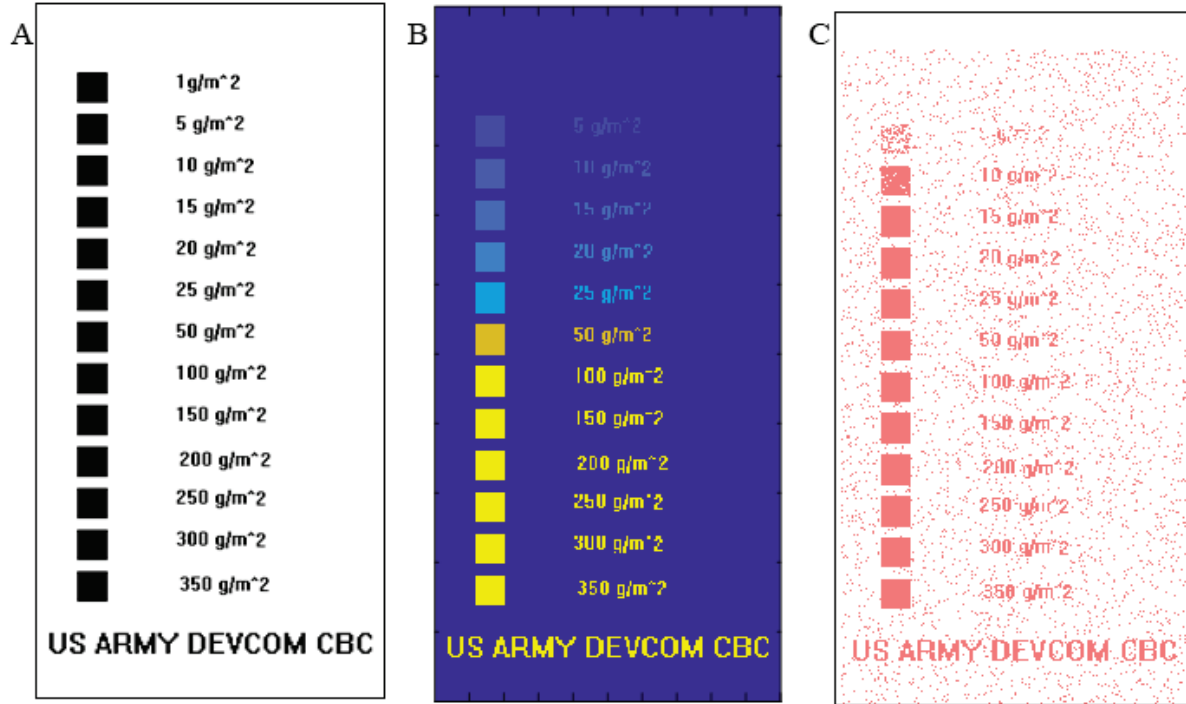


Figure 19: Binarized ground truth mask (A), a scaled mask showing the concentration gradient that is realized when the simulated spectra are applied (B), and the results of the anomaly detection algorithm showing a LOAD of 5g/m^2 (C).

3.5.2.1 Epsilon Optimization

Epsilon is one of the parameters determined within the algorithm that we employ, which is the ideal Euclidean distance for points in an area to be considered a cluster. While identifying the knee point of the k-distances graph (which effectively defines the point of diminishing returns for distance/cluster size), we hypothesized that it may be good to add a factor associated with that value to help tease out additional information. This is especially useful if there are very small differences between the background and the analyte that we seek to find anomalies from. We completed a study analyzing the LOAD of different analyte/background combinations at various concentrations using the ground truth mask and varied the coefficient factor associated with epsilon (epsilon factor).

The epsilon factor was varied from 0.001 to 100 in 11 increments. Through this work, we saw that increasing the epsilon factor increased the LOAD value. That is, at an epsilon factor of 100, many of the analyte/background pairs had a calculated LOAD of 200 g/m² compared to single or double digit LOADs at an epsilon factor of 1. Additionally, over 75% of the tested conditions resulted in no increase in performance (i.e. no improvement in the LOAD) when the epsilon factor was decreased below 1. Therefore, we determined that the ideal epsilon factor to use was a factor of 1, indicating that the ideal distance determined by the knee point calculation was ideal for *most* cases.

3.5.3 Non-Uniformity in Ground Truth Mask

One assumption that was made during the development of the ground truth is that there is uniform coverage over the analyzed area. This is true on the macro scale. However, through additional analysis it was realized that the area that is integrated with our detector results in non-uniform performance. Figure 20A shows a representative simulated dissemination that covers an area of 0.450m x 0.756m, which is the width that the sensor employed interrogates on the ground (0.756m) and the length of surface traversed at 5mph (0.450m). The red boxes represent an approximate area that a pixel would interrogate. We clearly see that the simulated droplets are sparse with respect to the total area interrogated by the sensor. Through 100 trials (i.e., different dissemination distributions) for each mass loading (0.1, 0.5, 1.0, 5.0, 7.5, 10.0g/m²) the percentage of simulated pixels that have a simulated droplet within their interrogated areas is presented in Figure 20B. This relationship can be applied to the mask in Figure 20C to identify pixels through random permutation at the appropriate percentage resulting in non-uniform or pixelated mask area, as seen in Figure 20D.

This finding is sensor, speed, and coverage dependent. Unfortunately, this was not fully studied during this work as it was realized at the end of the program. This is an important factor to investigate as the current work we have presented and will present the remainder of this report assumes uniform coverage. This suggests that our simulated LOADs and performance metrics will be liberal in their estimates. A re-analysis and of the LOADs associated with each analyte/background combinations is necessary for more accurate estimations of algorithm performance. However, generalized performance with respect to detection capabilities in the thresholds, VNIR versus SWIR, the RTS, and the actual performance of the algorithm in real time are still valid as these are not a direct result of the uniform coverage assumption.

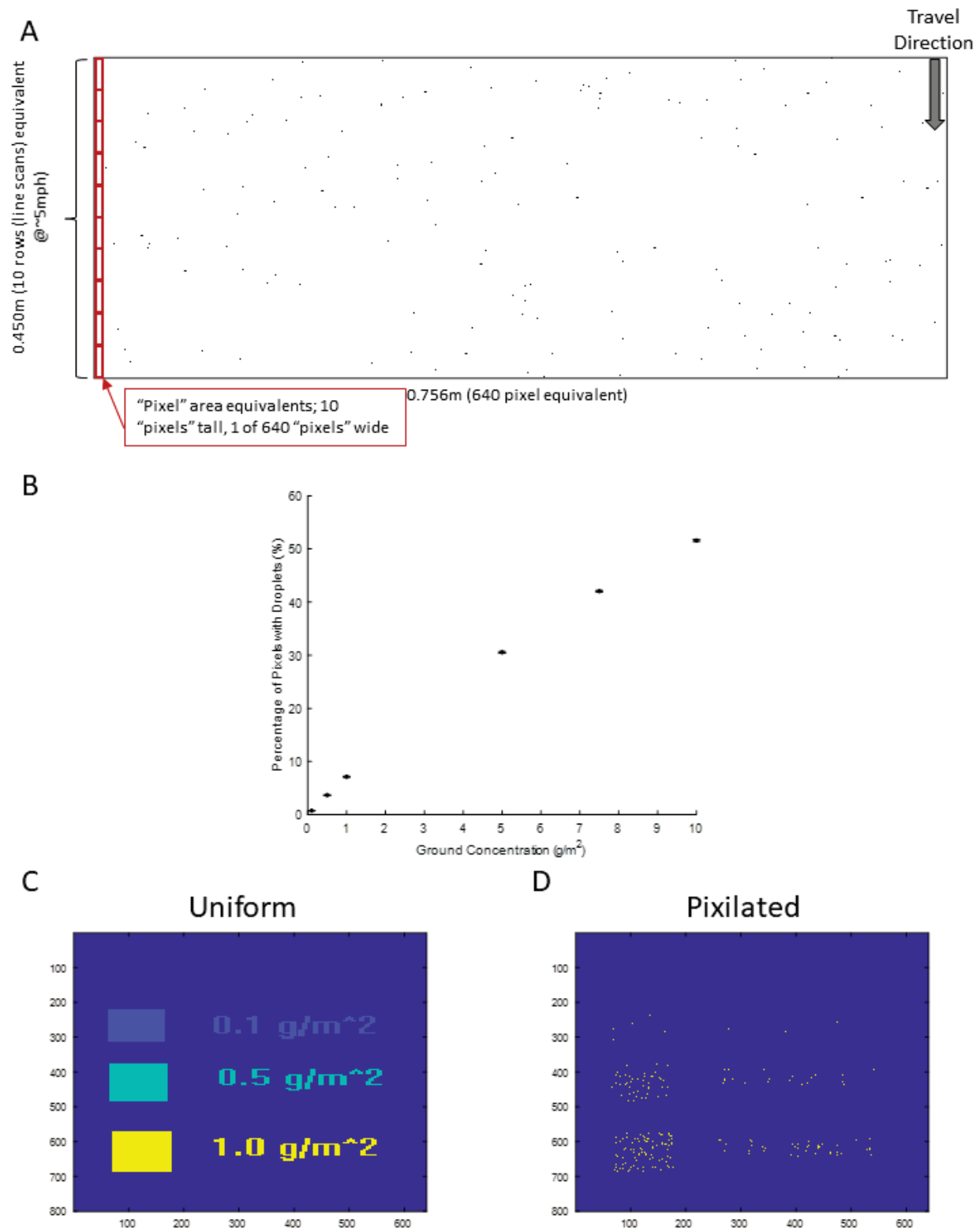


Figure 20: Simulated dissemination and a representation of the pixel area during travel at 5mph (red boxes, A), the percentage of pixels with droplets residing within the integrated area as a function of ground concentration (B), the uniform coverage of 0.1, 0.5, and 1g/m^2 , and the resulting coverage with non-uniform coverage.

3.6 Correction and Illumination Analysis

In understanding the best methodology for analysis with the data we receive, we investigated two additional factors. The first is if completing a radiometric correction within our analysis scheme was necessary prior to anomaly detection and the second was differences in solar illumination versus unnatural laboratory illumination.

To complete this analysis, indoor and outdoor data were taken on the same day with similar temperature and humidity levels to ensure appropriate comparisons between the data sets. These data were obtained within the same hour, about 30 minutes apart on 27 APR 2021 between 1430 and 1600 hours. Indoor conditions were set for 74°F at 50% relative humidity. Outdoor conditions were $73.7^{\circ}\text{F} \pm 1.3^{\circ}\text{F}$ with a relative humidity of $48.4\% \pm 2.2\%$, as seen in Figure 21. The temperature (black points) and relative humidity (red points) were comparable to indoor conditions. These data were provided by the Atmospheric Effects Branch on Aberdeen Proving Ground

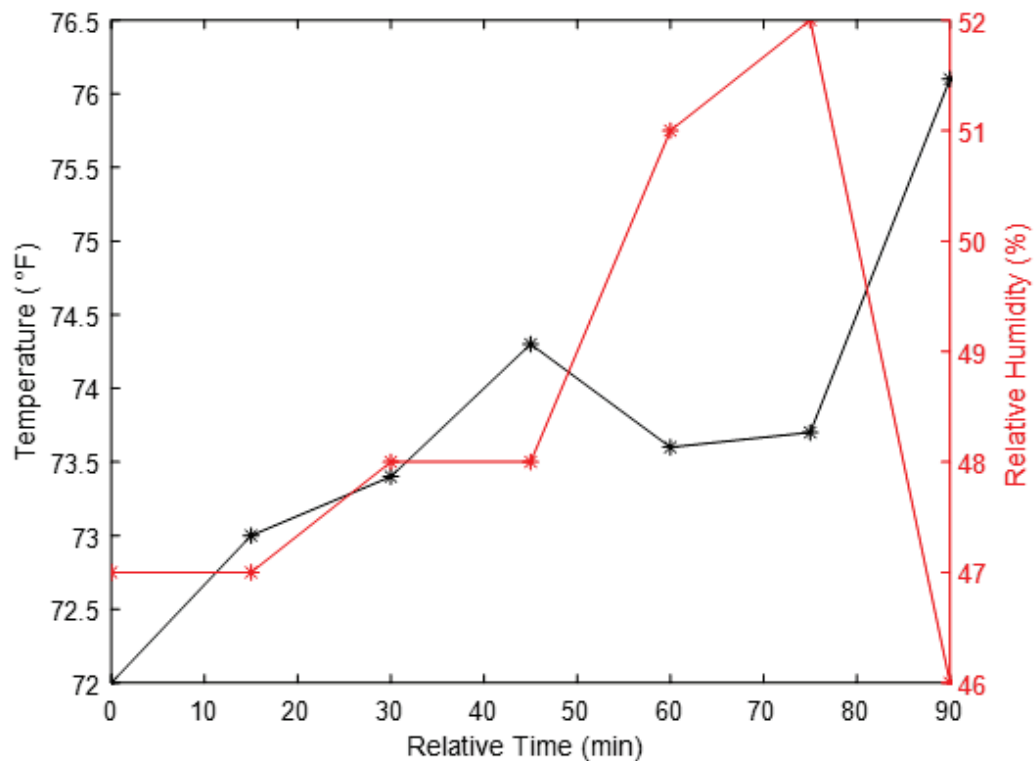


Figure 21: Temperature (black points) and humidity (red points) levels during the outdoor data acquisition.

3.6.1 Radiometric Corrections

A radiometric correction requires a dark acquisition which is completed by putting the lens cap on and taking an acquisition with the same parameters that will be taken during analysis. This poses several challenges for the Warfighter if this technology were to become fielded. The first is the use of a lens cap or electronic shutter, and the second would be to take this corrective measure. Since we are operating on specifically the statistical distributions

and parameters associated with these, we studied if a radiometric correction improved LOADs compared to if we didn't use a radiometric calibration.

Each analyte was studied independently with artificial illumination and solar illumination within each region that the detectors operated in. The data are shown in Table 2. Each analyte/background pair was aggregate together as a percentage of improvement. As an example, the analyte was paired with the backgrounds of AS, BC, BCP, G, RM, RCP, RD, and WCP; if ACE with WCP was the only analyte/background pair that produced a lowered LOAD, 12.5% of the analyte/background combinations would provide a lowered LOAD. The data show that for ACE and AN, radiometric correction either produced no change or made the measurements less sensitive for the majority (50% or more) analyte/background pairs. For PC, the solar illumination in the SWIR is the only category in which radiometric corrections produced a favorable change, on average. Maximum LOAD performance is more likely to be maintained without applying the radiometric correction as shown through the largely consistent data suggesting that there is, on average, no improvement or a worsening of the LOADs with the radiometric correction.

Table 2: Change in LOAD With Radiometric Corrections

Analyte	Illumination	Wavelength Region	LOAD Improvement (%)	LOAD Deterioration (%)
ACE	Artificial	VNIR	12.5	37.5
		SWIR	12.5	37.5
	Solar	VNIR	0	75
		SWIR	50	37.5
AN	Artificial	VNIR	25	50
		SWIR	0	0
	Solar	VNIR	25	37.5
		SWIR	0	0
PC	Artificial	VNIR	0	0
		SWIR	37.5	25
	Solar	VNIR	0	0
		SWIR	62.5	0

3.6.2 Illumination Analysis

To better ascertain the versatility of this technique in both artificial and natural (solar) illumination, a theoretical study was completed to see if there was a difference between the LOAD when using artificial or solar illumination sources. The data in Table 3 show that for each analyte, there is no average improvement in data acquired with solar illumination over artificial illumination. For the majority of the analytes and wavelength regions there is a higher proportion of the analyte/background pairs showing a deterioration in the LOAD or no change in the LOAD. The one exception is in the VNIR with ACE whereby 50% of the analyte/background pairs showed improvement with solar illumination. This is suggestive that limiting this technique to artificial illumination or solar illumination is not necessary. Since there is no consistent trend with improvement or deterioration between solar or artificial illumination, the authors believe this technique can be used in either situation making this a lighting agnostic technique.

Table 3: Change in LOAD with Solar Illumination

Analyte	Wavelength Region	LOAD Improvement ^a (%)	LOAD Deterioration ^b (%)	No Change (%)
ACE	VNIR	50	25	25
	SWIR	37.5	50	12.5
AN	VNIR	25	50	25
	SWIR	43	0	57
PC	VNIR	25	37.5	37.5
	SWIR	25	37.5	37.5

^a Improvement with respect to moving from indoor illumination to outdoor illumination

^b Deterioration with respect to moving from indoor illumination to outdoor illumination

3.7 Improvements Through Programing Languages

The original algorithm and work has been completed in MATLAB 2019b. One potential method of improving timing associated with processes is through moving from MATLAB to a different programming language, such as Python providing up to 500x improvement in computation time.⁴⁰⁻⁴¹ Other methods could be through pipelining the program to optimize the way that the program is run.⁴² This was investigated by converting the original MATLAB code to Python.

While there were necessary changes to the structure of program due to differences in the programming languages, the core algorithm remained the same. The pixel-wise anomaly detection differences are shown in Figure 22A with the Python-only identified anomalies in pink and the MATLAB-only identified anomalies in yellow. This shows the difference between the two anomaly algorithms. Interestingly, the anomalies that were identified in the Python code are shadows in the data due to two lamp posts and trees. The pixels that were identified as an anomaly that were common between the two implementations of the algorithm are shown in Figure 22B. A summary of the results is shown in Figure 22C with python identifying 2.6% less anomalous pixels than MATLAB, but shows an improvement of 57 times.

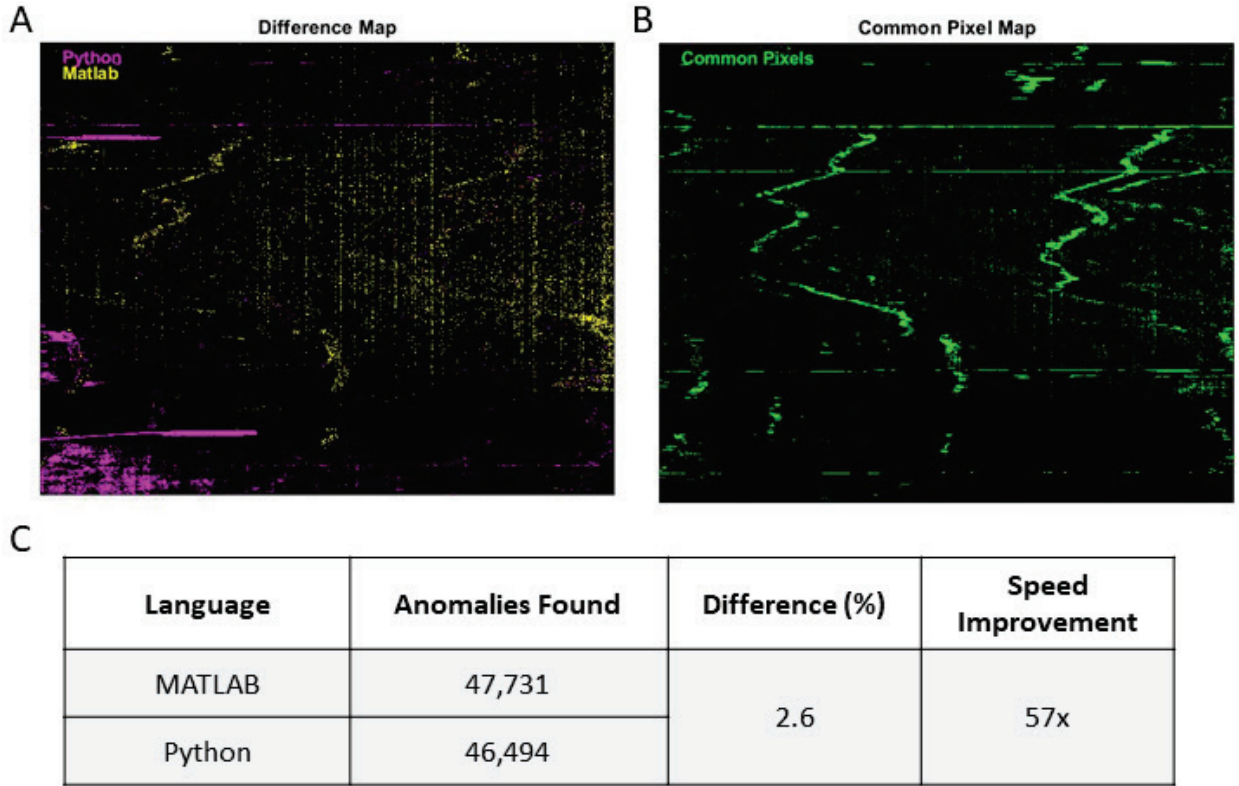


Figure 22: Differences in anomaly detections between implementing our anomaly in MATLAB and Python. A difference map showing the pixels that were anomalous with implementation of Python in pink or MATLAB in yellow (A), common pixels both identified as anomalies in both Python and MATLAB (B), and the speed improvement associated with implementation in Python (C).

The findings provided in Figure 15 and Figure 22 show that there are multiple ways to increase the processing time associated with our algorithm. These are utilizing a single detector that provides more sensitive response to anomalies, thereby bringing the individual line-read time below the 100ms threshold for real time analysis. Alternatively, we could realize up to 57x improvement in our timings through fully moving from MATLAB to Python. This has the theoretical improvement to a single line read to approximately 3ms per line.

4. CONCLUSIONS

Through the work presented here in Figure 5, Figure 6, Figure 7, Figure 8, and Figure 9, we show that we can properly model and evaluate the parameters used within our anomaly detection algorithm. In addition to modeling, we develop a method to threshold the data, moving towards updatable parameters while running the algorithm. In Figure 10 and Figure 11 we show the utility in parametrizing the data into the HOS and entropy and how we are then able to separate out the data in three dimensions and identify which parameter and regions are most easy to differentiate. We show our ability to theoretically and experimentally achieve real time anomaly detection at a rate of 100ms or less per detector and per frame in Figure 12, Figure 13, Figure 14, Figure 15, and Figure 16. Our algorithm's capabilities and limitations are highlighted through the use of ground truth data sets which provide us a statistical measure and

figure of merit to discuss the performance of the algorithm in Figure 17, Figure 18, Figure 19, Figure 20, and Figure 21 along with the data presented in Table 2 and Table 3. Lastly, we show how moving from MATLAB to Python provides a theoretical improvement in our algorithm processing speed in Figure 22, though this has not been validated with experimental data yet.

A main goal of this work was to develop and validate a real time anomaly detection method for the detection of explosives-related compounds. Through the work shown above, we achieved real-time anomaly detection of explosive-related and pharmaceutical-related compounds. While there is still room for improvement, we have shown that we can provide real time anomaly detection for a single detector with sub 100ms frame rate.

LITERATURE CITED

1. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Computing Survey*. **2009**, *41* (3), pp 1–58.
2. Wickens, T.D. *Elementary signal detection theory*. Oxford University Press, 2002.
3. Stanislaw, H.; Todorov, N. Calculation of signal detection theory measures. *Behavior research methods, instruments, & computers*. **1999**, *31* (1), pp 137–149.
4. Markou, M.; Singh, S. Novelty detection: a review—part 1: statistical approaches. *Signal processing*. **2003**, *83* (12), pp 2481–2497.
5. Markou, M.; Singh, S. Novelty detection: a review—part 2: neural network based approaches. *Signal processing*. **2003**, *83* (12), pp 2499–2521.
6. Li, W.; Du, Q. Collaborative representation for hyperspectral anomaly detection. *IEEE Trans. Geosci. Electron.* **2014**, *53* (3), pp 1463–1474.
7. Schaum, A. Hyperspectral anomaly detection beyond RX. In *Proceedings Volume 6565, Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery*. SPIE Defense + Security XIII, **2007**; p 656502.
8. Burger, J.; Gowen, A. Data handling in hyperspectral image analysis. *Chemom. Intell. Lab. Syst.* **2011**, *108* (1), pp 13–22.
9. Chang, C.-I. *Hyperspectral data processing: algorithm design and analysis*. John Wiley & Sons: **2013**.
10. Manley, P. V.; Sagan, V.; Fritschi, F. B.; Burken, J. G. Remote Sensing of Explosives-Induced Stress in Plants: Hyperspectral Imaging Analysis for Remote Detection of Unexploded Threats. *Remote Sensing*. **2019**, *11* (15), p 1827.
11. Lees, H.; Zapata, F.; Vaher, M.; García-Ruiz, C. Simple multispectral imaging approach for determining the transfer of explosive residues in consecutive fingerprints. *Talanta*. **2018**, *184*, pp 437–445.
12. Onat, B. M.; Carver, G.; Itzler, M. A solid-state hyperspectral imager for real-time standoff explosives detection using shortwave infrared imaging. In *Proceedings Volume 7310: Non-Intrusive Inspection Technologies II*; SPIE: Defense, Security, and Sensing. **2009**, 731004.
13. del Águila, A.; Efremenko, D.; Trautmann, T. A review of dimensionality reduction techniques for processing hyper-spectral optical signal. *Light and Engineering*. **2019**, *27* (3), pp 85–98.
14. Geng, X.; Sun, K.; Ji, L.; Tang, H.; Zhao, Y. Joint Skewness and Its Application in Unsupervised Band Selection for Small Target Detection. *Scientific Reports*. **2015**, *5*, p 9915.
15. Hsieh, P.-F.; Landgrebe, D. Statistics enhancement in hyperspectral data analysis using spectral-spatial labeling, the EM algorithm, and the leave-one-out covariance estimator In *Proceedings Volume 3438*. SPIE International Symposium on Optical Science, Engineering, and Instrumentation. **1998**.
16. Mendel, J. M. Tutorial on higher-order statistics (spectra) in signal processing and system theory: theoretical results and some applications. *IEEE*. **1991**, *79* (3), pp 278–305.
17. Sanaullah, M. A Review of Higher Order Statistics and Spectra in Communication Systems. *Global Journal of Science Frontier Research*. **2013**.

18. Wenke, L.; Dong, X. Information-theoretic measures for anomaly detection. In *Proceedings IEEE Symposium on Security and Privacy*. S&P **2001**, 14–16 May 2000; pp 130–143.
19. Marcum, J. A statistical theory of target detection by pulsed radar. *IRE Trans. Inf. Theory*. **1960**, 6 (2), pp 59–267.
20. Akaike, H. In *Prediction and Entropy*, New York, NY, Springer New York: New York, NY, 1985; pp 1–24.
21. Barron, A. R. Entropy and the Central Limit Theorem. *The Annals of Probability*. **1986**, 14 (1), pp 336–342.
22. Languirand, E.; Emge, D. Higher order statistics for anomaly detection in hyper spectral imaging. In *Proceedings Volume 11010: Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XX*. SPIE Defense + Commercial Sensing, **2019**, p 110100J.
23. Languirand, E.R; Curtiss, J.M; Emge, D. Field capability of a COTS instrument for on the move anomaly detection. *Presented at the 2019 Joint MSS Battlespace Acoustic, Seismic, Magnetic, and Electric-Field Sensing and Signatures Committee (BAMS) and National Symposium on Sensor & Data Fusion Committee (NSSDF) Conference*. Military Sensing Symposium, San Diego, CA, October 21–24 2019.
24. Jones, C.; Pewsey, A. The sinh-arcsinh normal distribution. *Significance*. **2019**, 16 (2), pp 6–7.
25. Jones, M. C.; Pewsey, A. Sinh-arcsinh distributions. *Biometrika*. **2009**, 96 (4), pp 761–780.
26. Rubio, F. J.; Ogundimu, E. O.; Hutton, J. L. On modelling asymmetric data using two-piece distributions. *Brazilian Journal of Probability and Statistics*. **2016**, 30 (3), pp 485–501.
27. Rosenblatt, M. A central limit theorem and a strong mixing condition. In *Proceedings of the National Academy of Sciences of the United States of America* **1956**, 42 (1), p 43.
28. Fischer, H. *A history of the central limit theorem: from classical to modern probability theory*. Springer Science & Business Media: 2010.
29. Lopes, R.; Reid, I.; Hobson, P. The two-dimensional Kolmogorov-Smirnov test. *XI International Workshop on Advanced Computing and Analysis Techniques in Physics Research*. **2007**.
30. Massey Jr, F. J. The Kolmogorov-Smirnov test for goodness of fit. *J. Am. Stat. Assoc.* **1951**, 46 (253), pp 68–78.
31. Van Campenhout, J.; Cover, T. Maximum entropy and conditional probability. *IEEE Trans. Inf. Theory*. **1981**, 27 (4), pp 483–489.
32. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*. **1979**, 9 (1), pp 62–66.
33. Gajjar, S.; Kulahci, M.; Palazoglu, A. Least Squares Sparse Principal Component Analysis and Parallel Coordinates for Real-Time Process Monitoring. *Industrial & Engineering Chemistry Research*. **2020**, 59 (35), pp 15656–15670.
34. Inselberg, A.; Reif, M.; Chomut, T. Convexity algorithms in parallel coordinates. *J. ACM*. **1987**, 34 (4), pp 765–801.

35. Kale, K. V.; Solankar, M. M.; Nalawade, D. B.; Dhumal, R. K.; Gite, H. R. A Research Review on Hyperspectral Data Processing and Analysis Algorithms. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences* **2017**, 87 (4), pp 541–555.
36. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H. P.; Xu, X. DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Trans. Database Syst.* **2017**, 42 (3), pp 1-21.
37. Languirand, E.R.; Emge, D.K. Real-time analysis of hyperspectral data in MATLAB: theoretical limits of anomaly detection utilizing higher order statistics through simulation; In *Proceedings Volume 11749: Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XXII*. SPIE Defense + Commercial Sensing, **2019**, p 117490Q
38. Brown, C. D.; Davis, H. T. Receiver operating characteristics curves and related decision measures: A tutorial. *Chemom. Intell. Lab. Syst.* **2006**, 80 (1), pp 24–38.
39. Long, G. L.; Winefordner, J. D. Limit of detection. A closer look at the IUPAC definition. *Anal Chem.* **1983**, 55 (7), pp 712A–724A.
40. Andrews, T. Computation time comparison between Matlab and C++ using launch windows. **2012**.
41. Matuska, S.; Hudec, R.; Benco, M. The comparison of CPU time consumption for image processing algorithm in Matlab and OpenCV. *2012 ELEKTRO*. **2012**, pp 75–78.
42. Hahn, T. T.; Humphreys, J. In *Demystifying digital signal processing (DSP) programming: The ease in realizing implementations with TI DSPs*, **2015**.

Blank

ACRONYMS AND ABBREVIATIONS

AA	aged asphalt
ACE	acetaminophen
AN	ammonia nitrate
AS	asphalt
BC	brushed concrete
BCP	black car panel
CBC	Chemical Biological Center
CDF	cumulative density function
CLT	central limit theorem
DBSCAN	density-based spatial clustering of applications with noise
DEVCOM	Combat Capabilities Development Command
FN	false negative
FP	false positive
FPR	false positive rate
G	gravel
GA	glucosamine
GR	grass
HOS	higher order statistics
KS	Kolmogorov-Smirnov
LOAD	limit of anomaly detection
ML	machine learning
PC	potassium perchlorate
PDF	probability density function
PL	parking lot
RAM	random access memory
RCP	red car panel
RD	red dirt
RM	rusted metal
ROC	receive operator characteristic
RTS	real time simulator
S	sucrose
SBC	sodium bicarbonate
SC	smooth concrete
SDK	software development kit
SWIR	short wave infrared
T	truvia
TN	true negative
TP	true positive
UAS	unmanned aerial system
VNIR	visible-near infrared
WCP	white car panel
WX	white clay

DISTRIBUTION LIST

The following individuals and organizations were provided with one electronic version of this report:

U.S. Combat Capabilities Development
Command Chemical Biological Center
(DEVCOM CBC)
FCDD-CBR-CP

ATTN: Languirand, E.

Emge, D.

Curtiss, J.

Ellzy, M.

DEVCOM CBC Technical Library

FCDD-CBR-L

ATTN: Foppiano, S.

Stein, J.

Defense Technical Information Center

ATTN: DTIC OA

Defense Threat Reduction Agency

DTRA-RD-IAR

ATTN: Pate, B.



U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND
CHEMICAL BIOLOGICAL CENTER