# A Novel Implementation of a Hyperspectral Anomaly Detection Algorithm For Real Time Applications With Pushbroom Sensors

**Conference Paper** · June 2019

DOI: 10.1109/WHISPERS.2018.8747221



Pablo Horstrand
Universidad de Las Palmas de Gran Canaria



Sebastian Lopez
Universidad de Las Palmas de Gran Canaria

# A NOVEL IMPLEMENTATION OF A HYPERSPECTRAL ANOMALY DETECTION ALGORITHM FOR REAL TIME APPLICATIONS WITH PUSHBROOM SENSORS

*Pablo Horstrand, Sebastián López, José Fco. López*

Institute for Applied Microelectronics (IUMA), University of Las Palmas de Gran Canaria (ULPGC)

## ABSTRACT

Anomaly detection is an increasingly important task when dealing with hyperspectral images in order to distinguish rare objects whose spectral characteristics substantially deviates from those of the neighboring materials. In this paper, a novel technique for accurate detection of anomalies in hyperspectral images is introduced. One of the main features of this method is its ability to process pushbroom data on-the-fly (i.e., line-by-line), being clearly suitable for real time applications in which memory resources are restricted as there is no need to store the whole hypercube. Diverse quality metrics have been applied on testing with real and synthetic hyperspectral data sets in order to compare the accuracy of the proposed algorithm over the state-of-the-art, showing the goodness of our proposal.

***Index Terms*—** Anomaly detection, push-broom sensors, real-time

## 1. INTRODUCTION

Anomaly detection (AD) is an important technique in hyperspectral data analysis that permits to distinguish rare objects with unknown spectral signatures that are not particularly abundant in a scene. This possibility of distinguishing a scarce group of pixels whose spectral signature significantly differs from their surroundings represents a crucial feature for different military and civilian applications, inspiring the appearance of a huge amount of hyperspectral AD algorithms in the recent scientific community.

The well-known ReedXiaoli (RX) algorithm [1] is one of the first developments in this field, being viewed as a benchmark to which other methods are compared. The RX anomaly detector is based on the Mahalanobis distance between the pixel under test and the background class. It assumes that the background follows a single Gaussian normal distribution and the probability density function is used to classify pixels as part of the background class. Thus, the background mean and inverse covariance matrix must be well estimated; otherwise, they could be contaminated by anomalies causing a subsequent misclassification. Several variations of the RX detection technique have been proposed in the literature in order to improve its performance. Subspace RX (SSRX) [2] and RX after orthogonal subspace projection (OSPRX) [3] are global anomaly detectors that apply principal component analysis (PCA) or singular value decomposition (SVD) to the datacube. The goal is to reduce the data volume to a smaller subspace where the first PCA/SVD bands are supposed to represent the background class. SSRX discards these bands, and then, RX is applied to the remaining subspace. On the contrary, OSPRX projects the data onto the orthogonal subspace before applying RX.

Unfortunately, all these algorithms require the sensing of the whole hyperspectral image before starting with the process of finding the anomalies in the captured scene. However, the most widely used sensors in nowadays remote sensing applications are based on pushbroom hyperspectral scanners, in which the image is captured in a line-by-line fashion, since they provide an excellent spectral resolution and take advantage of the movement of the aircraft or satellite that carries them for capturing the whole hypercube. Hence, for applications under strict real-time constraints in which the captured images must be processed in a short period of time, it is smarter and much more efficient if the anomalies are uncovered as soon as the hyperspectral data are sensed. Moreover, this kind of on-the-fly anomaly detection drastically reduces the formidable amount of memory that is required on-board the sensing platform in order to store the entire hypercubes. In this scenario, this paper proposes a novel hyperspectral anomaly detection algorithm specially conceived for being able to process the lines of pixels captured by a pushbroom scanner as soon as they are sensed. More concretely, the proposed Line-by-Line Anomaly Detection (LbL-AD) algorithm is based, as the OSPRX algorithm, on the concept of orthogonal subspace projections but employing a processing chain that guarantees a precise detection of the anomalies present in a hyperspectral scene with a low computational budget.

The rest of the paper is organized as follows. Section 2 describes, step by step, the proposed LbL-AD algorithm. Section 3 presents the hyperspectral data and the assessment metrics utilized for comparing the performance given by the LbL-AD algorithm versus other state-of-the-art proposals and

This work has been supported by the European Commission through the ECSEL Joint Undertaking (ENABLE-S3 project, No. 692455) and the Spanish Goverment through the projects ENABLE-S3 (No. PCIN-2015-225) and PLATINO (No. TEC2017-86722-C4-1-R).

outlines the main results obtained. Finally, Section 4 draws the most representative conclusions achieved in this work.

## 2. PROPOSED ALGORITHM

In order to keep a low computational complexity of the anomaly detection process, the proposed LbL-AD algorithm follows a twofold strategy: it starts processing as a whole bunch the first $n$ lines captured and then it follows a progressive line-by-line processing for the rest of the lines of the image in which only a reduced amount of operations are performed in order to update the results for the new acquired line of pixels. In particular, the following procedure is carried out for the first $n$ lines captured by the sensor:

1. A reference mean value is computed considering all the pixels contained in these lines.
2. The previously obtained mean value is subtracted to each sensed pixel.
3. The covariance matrix corresponding to these pixels is calculated. At this point, it is worth to highlight that this covariance matrix is not divided by the total number of pixels, since in this way this matrix can be reutilized in the next iterations (next lines of pixels), adding to it the new covariance matrix of the acquired line. In order for this to work, the number of pixels so far processed needs to be accounted as well.
4. Single value decomposition is performed then onto the covariance matrix to find the $d$ highest eigenvalues and their associated eigenvectors. Here, in order to keep the LbL-AD algorithm within a low computational burden, the following computing strategy is applied:
   a) First, the highest eigenvalue and its associated eigenvector are obtained by means of the power iteration method [4][5].
   b) Afterwards, deflation [6] is performed onto the covariance matrix to calculate the next $d - 1$ eigenvalues and eigenvectors, by means of successively applying the power iteration method. This process is repeated until the $d$ desired number of components is obtained.

   The reason for having selected this combination of methods is dual: on one hand, it is a very fast method for just obtaining a few principal components, and on the other hand, if the algorithm is wisely initialized, the computation time can be significantly reduced. This last characteristic is exploited by the proposed LbL-AD algorithm, as for each iteration (each new line of pixels that is acquired after the first $n$ lines) the subspace calculated from the previously sensed pixels is used for initializing the algorithm, which brings a significant speedup factor to the process.
5. The pixels are projected onto the subspace spanned by the $d$ eigenvectors obtained in the previous step.
6. The Mahalanobis distance is calculated for each pixel, as it is done in the original RX algorithm. This step involves the calculation of the pseudoinverse of the covariance matrix. However, as far as the covariance matrix in the new projected subspace is a diagonal matrix, its inverse is obtained by just inverting its single elements individually, which is again a huge save in terms of computing time.
7. Based on the calculated distance result, it is decided whether each pixel is an anomaly or not.

Once these $n$ lines of pixels have been processed the following line-by-line procedure is carried out for the rest of the pixels in the hyperspectral scene under analysis:

A. The mean value obtained in step 1 is subtracted to each pixel of the line under processing.

B. The covariance matrix of the line under processing is calculated and added to the existing covariance matrix calculated previously in step 3. As with step A, this is performed under the assumption that the mean value obtained in step 1 remains approximately constant, which allows to skip the full computation of a new covariance matrix for each new line of pixels. Moreover, this method allows us to keep in memory just the previous covariance matrix, but not the entire amount of pixels processed, which means a huge save in memory space and access time.

C. Steps 4 to 7 are applied to the line of pixels under processing.

Figure 1 shows a general vision of the stages involved in the algorithm. On the top we have a $m$ bands hyperspectral image (HSI) composed of $N$ lines, each of them with $lSize$ amount of pixels. The first $n$ lines are buffered into a matrix $M$ which afterwards is used to calculate the mean value $\mu$ to be substracted to every processed image line. The covariance $\Sigma$ matrix results from the obtained normalized matrix $M'$, for which the eigenvectors $\vec{u_1}, ..., \vec{u_d}$ and the eigenvalues $s_1, ..., s_d$ are calculated. $M'$ is then projected into the new subspace defined by the calculated set of eigenvectors in order to obtain matrix $M''$. Finally, the Mahalanobis distance is calculated for each pixel by using the covariance matrix in the new subspace, represented by a diagonal matrix formed by the eingenvalues in descending order. After the first $n$ lines have been processed, the proposed algorithm processes the sensed data line by line, as it is described in the right hand side of Figure1. As it has been already mentioned, the mean value $\mu$ obtained in the first stage of the algorithm is substracted from each pixel, and then, the covariance matrix is updated to obtain $\Sigma_{new}$ and the new set of $d$ eigenvalues and eigenvectors for that $\Sigma_{new}$ are calculated. The process then continues as before, i.e., projecting the image onto the new subspace and obtaining the Mahalanobis distance for each pixel.

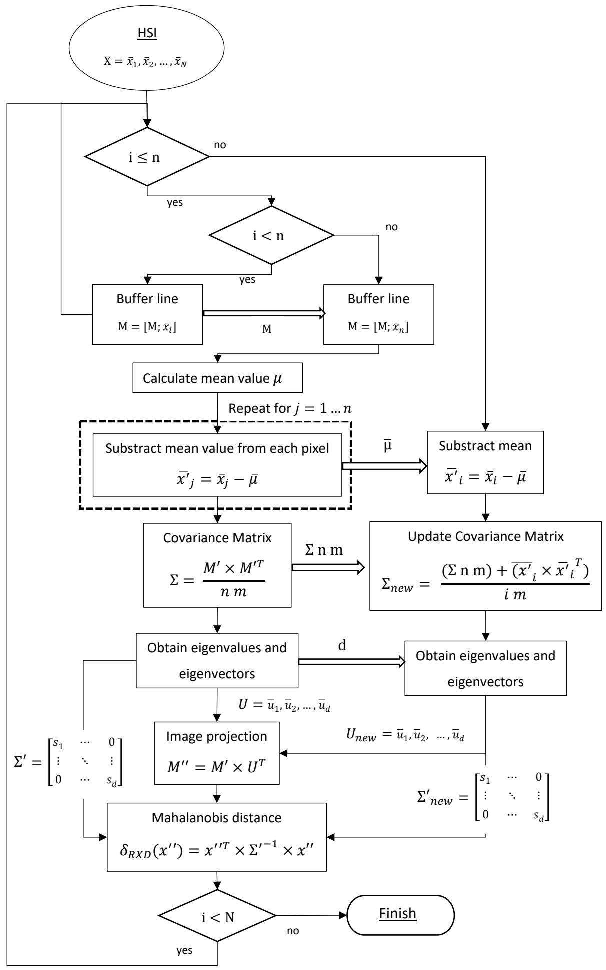



**Fig. 1**. General overview of the LbL-AD algorithm.

## 3. RESULTS

This section first introduces the hyperspectral data and the assessment metrics that have been used in order to demonstrate the efficiency of the proposed technique and then, the results obtained are presented and compared with those obtained with state-of-the-art algorithms.

### 3.1. Hyperspectral Data

The simulated data have a size of 150×150 pixel vectors and 429 spectral bands. They were generated using a spectral library collected from the United States Geological Survey (USGS) [7]. Background was simulated using four different spectral signatures whose abundances were generated using a Gaussian spherical distribution. Twenty panels of different sizes arranged in a 5×4 matrix were introduced as anomalies. There are five 4×4 pure-pixel panels lined up in five rows in the first column, five 2×2 mixed-pixel panels in the second column, five subpixel panels combined with the background in a proportion of 50% in the third column and five subpixel panels blended with the background in a proportion of 75%. Therefore, the simulated image has 110 anomaly pixels, a 0.49% of the image, as it is illustrated in Figure 2a. This data set is very challenging for AD because of the high spectral similarities between some anomalous and background signatures.

In order to test the proposed algorithm in a more realistic scenario, four real hyperspectral data sets have been additionally used. The first real data set was taken over the Rochester Institute of Technology (RIT) by the Wildfire Airborne Sensor Program (WASP) Imaging System [8]. The system covers the visible, short, mid and long-wave infrared regions of the spectrum. A portion of the overall image taken over a parking lot with a size of 180×180 pixels and 120 bands has been used in this study, as can be seen in Figure 2b, where anomalies are fabric targets which consist on 72 pixels and account for 0.22% of the image. The second real data set was collected by the NASA Jet Propulsion Laboratorys Airborne Visible Infra-Red Imaging Spectrometer (AVIRIS) over the World Trade Centre (WTC) area in New York City on September 16, 2001 [9]. The original data set has a size of 614×512 pixels and 224 spectral bands from 0.4 to 2.5 $\mu$m, although a smaller region with a size of 200×200 pixels was selected as data set. Anomalies are thermal hot spots which consist on 83 pixels and account for 0.21% of the image scene. Figure 2c shows a representation of this image.

Finally, a real HSI was generated by one of our cameras in our hyperspectral imaging laboratory using the Headwall hyperspectral sensor Hyperspec® which operates in the shortwave infrared range (SWIR) between 0.9-2.5 $\mu$m. It is a push-broom camera which provides 384 spatial bands and 273 spectral bands [10]. However, due to low-signal-to-noise ratio (SNR) of the first and last spectral bands, they were removed (1-4, 269-273), so that, 264 available bands were retained. The image scene covers an area of 199×170 pixels as shown in Figure 2d. Anomaly targets are some legumes and two plastic squares and they consist of 323 pixels, 0.95% of the image. The main background material is beach sand.

### 3.2. Assessment Metrics

In order to compare the efficiency of the different detection algorithms in terms of the accuracy of their detection results, Receiver Operating Characteristic (ROC) curves and the area under these curves (AUC) have been widely used in the literature. ROC curves are two dimensional graphical plots which illustrate the relation between the true positive rates (TPR) and the false positive rates (FPR) obtained for various thresh-

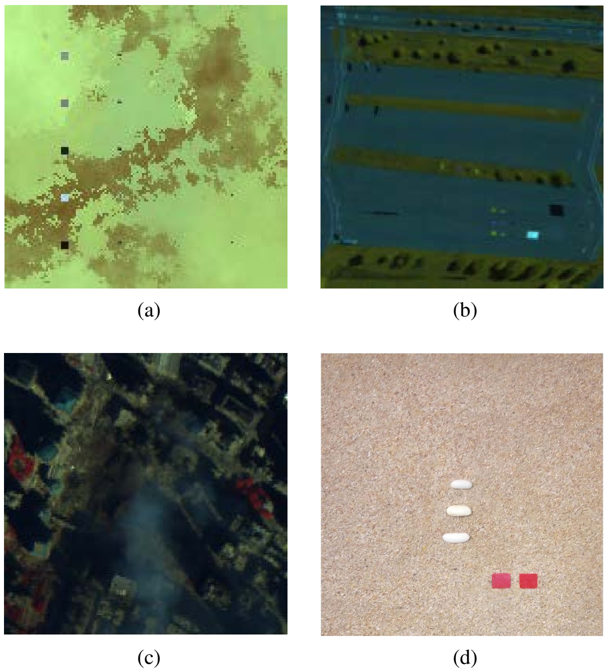


**Fig. 2**. Test data sets. (a) Synthetic image. (b) WASP RIT scene. (c) AVIRIS WTC scene. (d) SWIR scene. Sand beach background.

old settings. To compare the performance of several AD algorithms, AUC is used as a scalar measure, so that, a representation with the biggest AUC outperforms the others. However, this metric does not always reflect how well the algorithm separates the anomalies from the background. For that reason, two extra quality metrics will be utilized in this work: Brier Score (BS) and Squared Error Ratio (SER)[11].

Brier Score (BS) measures the accuracy of probability predictions in terms of marking anomalies and background pixels with the highest and the lowest scores, respectively. If anomaly pixels are represented as *ones* and background pixels as *zeros* in the ground-truth, then, the BS for each type of pixel is calculated as:

$$\mathrm{BS}_{\mathrm{anomaly}} = (\mathbf{p}_i - 1)^2; \quad (1)$$

$$\mathrm{BS}_{\mathrm{background}} = (\mathbf{p}_i - 0)^2; \quad (2)$$

A global scalar metric named Squared Error Ratio (SER) is obtained dividing the sum of all squared differences by the total number of pixels in the image.

$$\mathrm{SER} = \frac{\sum_{i=1}^{n_{\mathrm{anomaly(GT)}}} (\mathbf{p}_i - 1)^2 + \sum_{j=1}^{n_{\mathrm{bck(GT)}}} (\mathbf{p}_j - 0)^2}{N_p} \cdot 100 \quad (3)$$

where $\mathrm{n}_{\mathrm{anomaly(GT)}}$ represents the number of anomalous pixels according to the ground-truth and $\mathrm{n}_{\mathrm{bck(GT)}}$ is the number of pixels belonging to the background class according to the ground-truth.

Furthermore, the first addend of the SER metric numerator is referred to as Anomaly_Error and it provides insight into the error made scoring anomalous pixels while the second addend, named Bck_Error, makes reference to the error scoring background pixels.

$$\mathrm{Anomaly\_Error} = \sum_{i=1}^{n_{\mathrm{anomaly(GT)}}} (\mathbf{p}_i - 1)^2 \quad (4)$$

$$\mathrm{Bck\_Error} = \sum_{j=1}^{n_{\mathrm{bck(GT)}}} (\mathbf{p}_j - 0)^2 \quad (5)$$

### 3.3. Detection Performance

In order to test the performance of the proposed algorithm, its detection efficiency has been compared with the OSPRX algorithm and with one recent state-of-the-art anomaly detector named PLP-KRXD [12] which is also able to process hyperspectral images in a line-by-line fashion. For this purpose, the LbL-AD algorithm parameters described in Section 2, $n$ and $d$, have been set to 10 and 5 respectively as these values were the ones that give the best average results with the test images considered in this paper. In this sense, it is also worth to mention that the PLP-KRXD has been initialized as reported by the authors in their paper.

The results obtained in terms of the metrics described in Section 3.2 are summarized in Table 1 for each of the already mentioned images. From these results, it is concluded that the proposed LbL-AD algorithm is able to provide similar results with respect to the OSPRX algorithm, with the major advantage of being able to compute the anomalies present in the image as soon as the line of pixels are sensed. In addition, it is also concluded from Table 1 that if we focus our attention only on the solution that allows to process the hyperspectral images in a line-by-line fashion, our proposal delivers a much better detection performance than the one provided by the PLP-KRXD.

Finally, Figure 3 illustrates the two-dimensional plots of the detection results given by the proposed LbL-AD algorithm for all the considered datasets, visually demonstrating that our proposal is able to capture in a very precise way all the anomalies present in a given hyperspectral image independently of the characteristics of the background and of the anomalies present in the scene under analysis.

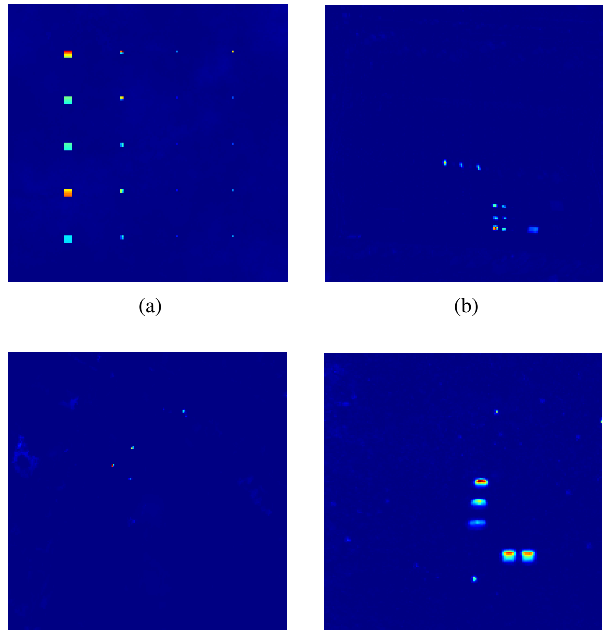
## 4. CONCLUSIONS

LbL-AD (Line-by-Line Anomaly Detection) algorithm is shown for the first time in this paper. This technique presents a substantial improvement with respect to other ones shown

**Table 1**. Assessment Metric Summary

| | AUC | Anomaly_Error | Bck_Error | SER |
|---|---|---|---|---|
| | Synthetic Image | | | |
| OSPRX | 1.000 | 48.7062 | 0.1587 | 0.2172 |
| LbL-AD | 1.000 | 34.1587 | 1.3211 | 0.1577 |
| PLP-KRXD | 0.9401 | 103.9872 | 8.0186 | 0.4978 |
| | WASP RIT scene | | | |
| OSPRX | 0.9994 | 40.9303 | 1.1583 | 0.1299 |
| LbL-AD | 0.9988 | 45.4502 | 1.4871 | 0.1445 |
| PLPK-RXD | 0.7829 | 71.9865 | 8.6668 | 0.2489 |
| | AVIRIS WTC | | | |
| OSPRX | 0.9727 | 72.9378 | 0.7676 | 0.1843 |
| LbL-AD | 0.9673 | 74.6396 | 1.3203 | 0.1899 |
| PLP-KRXD | 0.5161 | 82.9752 | 5.8713 | 0.2221 |
| | SWIR beach background | | | |
| OSPRX | 0.9996 | 130.3023 | 7.7523 | 0.4081 |
| LbL-AD | 0.9993 | 163.486 | 4.462 | 0.4965 |
| PLP-KRXD | 0.7988 | 322.8650 | 4.7671 | 0.9685 |

previously in the scientific literature, being capable of processing on-the-fly hyperspectral images captured by pushbroom sensors. Quality measurements have been applied with different real and artificial hyperspectral images, opening new avenues when real time applications and memory restrictions are the target.



(a) (b) (c) (d)

**Fig. 3**. Two-dimensional plots of the detection results obtained by the proposed LbL-AD algorithm for all data sets.

## 5. REFERENCES


[1] I. S. Reed and X. Yu, "Adaptive multiple-band cfar detection of an optical pattern with unknown spectral distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 10, pp. 1760–1770, 1990.

[2] A. Schaum, "Joint subspace detection of hyperspectral targets," in *Aerospace Conference, 2004. Proceedings. 2004 IEEE*, vol. 3. IEEE, 2004.

[3] C.-I. Chang, "Orthogonal subspace projection revisited," *Hyperspectral Data Processing: Algorithm Design and Analysis*, pp. 355–390, 2013.

[4] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *J. Res. Natl. Bur. Stand. B*, vol. 45, pp. 255–282, 1950.

[5] W. Shi and D. Zhang, "The power and deflation method based kernel principal component analysis," *2010 IEEE International Conference on Intelligent Computing and Intelligent Systems*, vol. 1, pp. 828–832, Oct 2010.

[6] R. B. Lehoucq and D. C. Sorensen, "Deflation techniques for an implicitly restarted arnoldi iteration," *SIAM Journal on Matrix Analysis and Applications*, vol. 17, no. 4, pp. 789–821, 1996.

[7] USGS digital spectral library. [Online]. Available: http://speclab.cr.usgs.gov/spectral-lib.html

[8] J. A. Herweg, J. P. Kerekes, O. Weatherbee, D. Messinger, J. van Aardt, E. Ientilucci, Z. Ninkov, J. Faulring, N. Raqueño, and J. Meola, "Spectir hyperspectral airborne rochester experiment data collection campaign," in *SPIE Defense, Security, and Sensing*. International Society for Optics and Photonics, 2012, pp. 839 028–839 028.

[9] A. Plaza, Q. Du, Y.-L. Chang, and R. L. King, "High performance computing for hyperspectral remote sensing," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, no. 3, pp. 528–544, 2011.

[10] *High-Efficiency Hyperspec® SWIR Imaging sensor for the 900nm to 2500nm spectral range*, Headwall Photonics, Inc, 10 2015.

[11] M. Diaz, R. Guerra, S. Lopez, and R. Sarmiento, "An algorithm for an accurate detection of anomalies in hyperspectral images with a low computational complexity," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 1159–1176, Feb 2018.

[12] C. Zhao, W. Deng, Y. Yan, and X. Yao, "Progressive line processing of kernel rx anomaly detection algorithm for hyperspectral imagery," *Sensors*, vol. 17, no. 8, 2017. [Online]. Available: http://www.mdpi.com/1424-8220/17/8/1815