Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 2 Issue 9, September - 2015

Fast Anomaly Detection Algorithms For Hyperspectral Images

J. Zhou
Google, Inc.
Mountain View, California, USA

C. Kwan
Signal Processing, Inc.
Rockville, Maryland, USA
chiman.kwan@signalpro.net

***Abstract*— Hyperspectral images have been used in anomaly and change detection applications such as search and rescue operations where it is critical to have fast detection. However, conventional Reed-Xiaoli (RX) algorithm [6] took about 600 seconds using a PC to finish the processing of an 800x1024 hyperspectral image with 10 bands. This is not acceptable for real-time applications. A more recent algorithm known as kernel RX (KRX) [7] achieves better detection performance than RX at the expense of computational cost. For example, for the same 800x1024 image with 10 bands, KRX took 15 hours to finish the processing. In this paper, we present a general framework for fast anomaly detection using RX and KRX algorithms. First, a fast data reduction scheme using Principal Component Analysis (PCA) is proposed. This method takes less than 1 second to finish and the performance degradation is minimal. Second, we propose several speed boosting options in the RX and KRX algorithms. These options include image sub-sampling, the use of block pixels, and background pixel sub-sampling. Actual hyperspectral image has been used in our studies. Receiver operating characteristics (ROC) curves and actual computation times were used to compare the various options. For the 800x1024x10 image, we were able to improve the speed by more than 220 times for RX and 700 times for KRX with minimal degradation in detection performance.**

Keywords—hyperspectral images; anomaly detection; RX; KRX; search and rescue.

I. INTRODUCTION

Hyperspectral images have gained popularity in recent years. NASA's Hyperion [1] has been operational since 2000. Moreover, NASA's AVIRIS imager [2] has been used for fire damage assessment in a number of places in the US for the past 2 decades. NASA is also planning the HyspIRI mission [3], which will cover the whole Earth. Comparing to color (GeoEye) [4] and multi-spectral imagers in LANDSAT [5] that have only 3-10 bands, hyperspectral imagers offer hundreds of spectral bands. As a result, the discrimination power using hyperspectral imagers are significantly better than multi-spectral counterparts.

The RX [6] algorithm has been widely used in many image processing applications. However, due to high dimensionality of hyperspectral images, the RX algorithm took about 600 seconds to finish the processing of an 800x1024 hyperspectral image with 10 bands. Recently, a kernel RX (KRX) algorithm was developed [7]. Although KRX has excellent performance in anomaly detection using hyperspectral images, one main issue is that it is computationally intensive. For example, we recently applied KRX to an 800x1024 hyperspectral image with 10 bands and it took about 15 hours to generate the detection results. This is not acceptable for real-time applications.

In this paper, we present a general framework for fast anomaly detection using RX and KRX algorithms. First, a fast data reduction scheme using Principal Component Analysis (PCA) is proposed. This method takes less than 1 second to finish and the performance degradation is minimal. Second, we propose several speed boosting options in the RX and KRX algorithms. These options include image sub-sampling, the use of block pixels, and background pixel sub-sampling. Actual data have been used in our studies. Receiver operating characteristics (ROC) curves and actual computation times were used to compare the various options. We were able to improve the speed by more than 220 times for RX and 700 times for KRX with minimal detection performance degradation.

Section II of the paper provides a brief description of the hyperspectral image used in this study. Section III describes the fast algorithms with detailed computational complexity analysis. Section IV summarizes the evaluation results. Section V provides the concluding comments and future work.

II. ABOUT HYPERSPECTRAL IMAGES

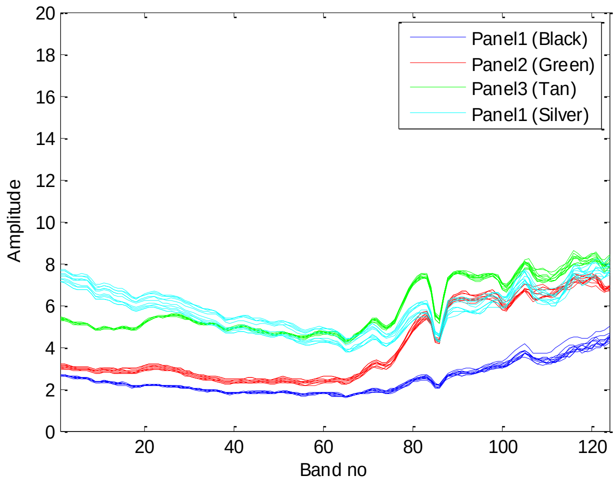
We have used a hyperspectral image from the US Air Force in this research for anomaly detection investigations. The image data was collected in October 2005. The collection instrument was a pan and tilt mounted VNIR Imaging Spectrometer with ~0.45 - 0.90 um wavelengths. For more information about the sensors, one should see [8]. This image has 2 small targets inserted in the form of tarp bundles.

Table 1 shows the five attributes of the hyperspectral images. These attributes are solar elevation (deg), solar azimuth (deg), date, collection time and weather conditions at the time of data collection. Fig. 1 shows the jpeg image of the AF image. The pixel signatures of the four panels are also shown in Fig. 1.

Table 1: Collection times and weather conditions for the AF image

Solar Elev. (deg)	Solar Az. (deg)	Date	Collection Time	Weather
41.80	176.29	10/14/05	13:11:05	Partly cloudy with Haze

(a) Oct 14 image with targets inside the circles.



(b) Four panel signatures in Oct 14 image

Fig. 1. Close look at the AF image with signature plots of the four panels in the images.

III. FAST ALGORITHMS

A. Complexity Analysis of RX and KRX

We first briefly review the RX and KRX algorithms in the following paragraphs. We then analyze their computational complexity.

Algorithm RX:
Input: Image *I*, inner window size *l*, outer window size *u*
Output: A single band image O which indicates the anomaly value
Algorithm:

For each pixel *p* in the inner window with signature $r = I(p)$;

Extract the background pixels $X_b = [x_1, x_2, \cdots, x_M]$ in the outer window;

Compute mean and covariance of X_b:

$$\mu_b = \frac{1}{M}\sum_{i=1}^{M} x_i$$

$$C_b = \frac{1}{M}\sum_{i=1}^{M} (x_i - \mu_b)(x_i - \mu_b)^T$$

Compute the Mahananobis distance as anomaly score

$$O(p) = (r - \mu_b)^T C_b^{-1} (r - \mu_b)$$

End

The background pixels are extracted as shown in Fig. 2.

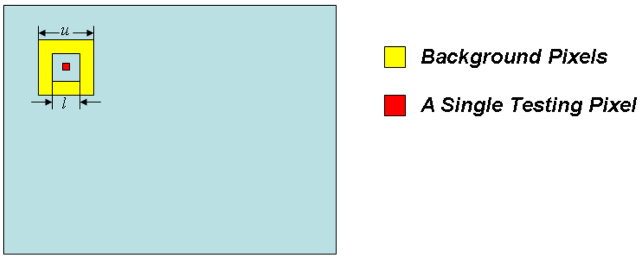


Fig. 2. Illustration of background pixel selection in RX algorithm.

The two core elements of RX algorithms are to:

1. Extract the background pixels X_b for a pixel r.
2. Compute an anomaly score *d* using a distance function $d = f(r, X_b)$

The RX algorithm assumes the background conforms to a normal distribution and uses the log likelihood in distance function. In contrast, the KRX algorithm first maps the pixels to a new feature space and then applies the RX algorithm, i.e.

$$f_{krx}(r, X_b) = f_{rx}(\phi(r), \phi(X_b))$$

where $f_{krx}(r, X_b)$ denotes the KRX distance function and $f_{rx}(r, X_b)$ denotes the RX distance function, $\phi(r)$ is a non-linear mapping of *r*. The dimension of the new feature space can be infinite. However, it is proven that f_{krx} can be directly calculated using *kernels* without explicit mapping. The kernels implicitly compute inner products in the feature space.

In RX algorithm, background extraction and distance calculation is done for every pixel. This iterative process makes the computation very expensive for an image with millions of pixels. The time complexity of a standard RX algorithm is *O*(*NC*), where *N* is the number of pixels and *C* is the time of distance calculation. Other RX based algorithms have the same form of complexity except that the *C* is different. For RX,

$$C_{rx} = O(MK^2) + O(K^3)$$

where *M* is the number of background pixels and *K* is the number of bands. The first term is the time complexity for covariance matrix computation and the second term is the time complexity for computing the inversion of covariance matrix, which is usually done using singular value decomposition (SVD). For KRX,

$$C_{krx} = O(M^2K) + O(M^3).$$

In general, $M >> K$. For instance, suppose the outer window size is 15, the inner window size is 5, and the band number is 10, then M = 200 and K = 10. Therefore, the time complexity can be simplified as

$$C_{rx} = O(MK^2) \text{ and } C_{krx} = O(M^3).$$

From the above analysis, we can see that KRX is much more expensive than RX. For instance, if M = 200 and K = 10, the time complexity of KRX algorithm is about 400 times of that of the RX algorithm.

As a matter of fact, even the RX algorithm cannot achieve real-time processing for high dimensional hyperspectral image. For instance, the Air Force hyperspectral image has 800 samples, 1024 lines and 124 bands. For outer window size of 45 and inner window size of 15, even only using 10 bands in the anomaly detection computations, it takes RX about 600 seconds to complete the process, which is not acceptable.

B. Fast Implementations

To reduce the computational complexity of RX and KRX, a straightforward approach is to first perform spectral band reduction by using Principal Component Analysis (PCA). In this work, we applied PCA to reduce the number of bands from 124 down to 10 bands. PCA can be finished in 1 second, which introduces minimal additional computational cost. After that, a dimension reduction in the spatial domain using *sub-sampling* is performed. Suppose the number of samples and lines are reduced by s times; the number of outer window size and inner window size are also reduced by s times; the number of bands are reduced by d times. Then the number of pixels N is reduced by s^2 times, the number of bands K is reduced by d times and the number of background pixels M is reduced by s^2 times. Suppose the background pixels are further sub-sampled such that the number of window dimension is reduced by c times. Then the overall speed boosting for RX algorithm is

$$B_{rx} = s^4d^2c^2$$

The speed boosting for KRX algorithm is

$$B_{krx} = s^8c^6$$

If s = 3, d = 10 and c = 2, then $B_{rx} = 32,400$ and $B_{krx} = 419,904$. Obviously, this is a significant reduction in computation time. However, even using these settings, both RX and KRX are still not real-time.

To further reduce the computational complexity, we proposed a concept called *block* pixel. Instead of using a unique background for each single pixel, this method uses the same background for a block of pixels to compute the distances, as Fig. 3 shows. Assuming the block size is b x b, a further b^2 time reduction can be achieved for both RX and KRX.

While the above complexity reduction techniques can significantly speed up the process, they are all approximation approaches and may degrade the performance. Fortunately, after applying a filtering on the detection results, the performance can be significantly improved and comparable to the results using original data.

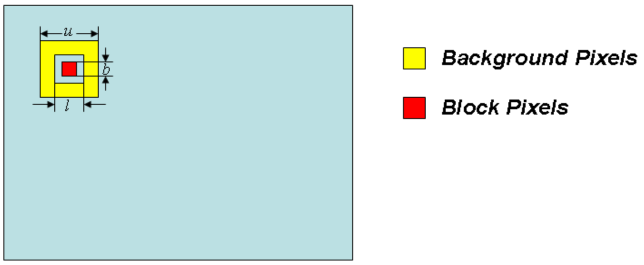


Fig. 3. Illustration of using the same background for a block of pixels.

IV. EVALUATION RESULTS

The complexity analysis in Section III gives order of magnitude estimates of the computational effort of various algorithms. In practice, due to other factors in the PC (multi-tasking, multiple processes, memory access, etc.), there will be differences between the estimates and the actual computation time. The experiments in this section were carried out on a sub-sampled (3x3) Air Force image (Fig. 1) where PCA was applied to reduce the number of spectral bands from 124 to 10. Before the application of 3x3 sub-sampling, the computational times for RX and KRX were 598 seconds and 15 hours, respectively, for the 800x1024x10 hyperspectral image.

Now, Fig. 4 and Fig. 5 show the ROC and computation time results for RX and KRX, respectively, where band number reduction and 3x3 sub-sampling have been implemented. Description of the labels in Fig. 4 and Fig. 5 are as follows:

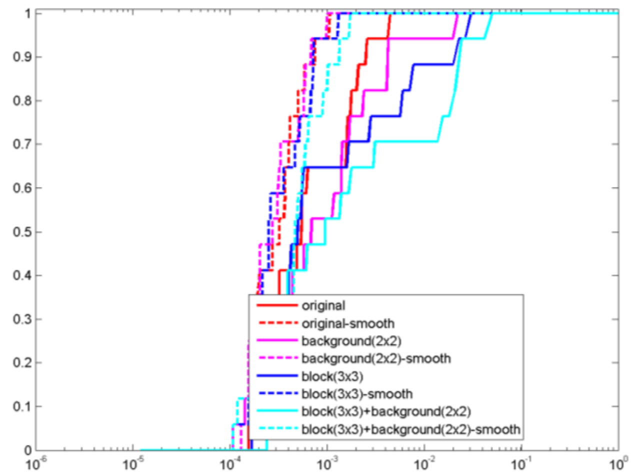
1. **original:** no background sub-sampling and block pixels options
2. **background (2x2):** 2x2 sub-sampling on background pixels
3. **block (3x3):** the block size is 3x3
4. **block(3x3)+background (2x2):** both block and background sub-sampling options are used
5. "**smooth**" means the anomaly detection result is smoothed using a 3x3 average filter.

RX results are shown in Fig. 4. We have the following observations:

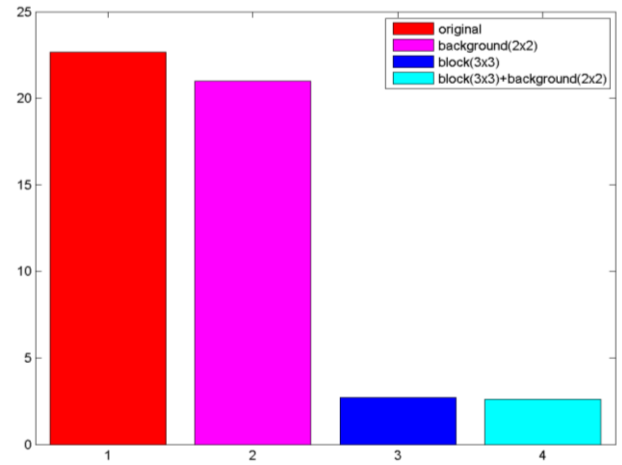
- Background sub-sampling does not bring much speed enhancement (Fig. 4b);
- Block pixel brings significant speed enhancement (Fig. 4b);
- After smoothing on the detection results, the anomaly detection performance of all

methods is similar (Fig. 4a), meaning that fast algorithms do not degrade the detection performance.

Note that the processing time of RX with all of the options is just about 2.7 seconds, which is 1/8 of the processing time (23 seconds) of the RX on the same sub-sampled image and is 1/220 of the original RX on the whole image (596 seconds) of 800x1024 image with 10 bands. If one compares with the computational time without sub-sampling (596 seconds) and with all options (2.7 seconds), then the speed improvement factor is 220.



(a) ROC



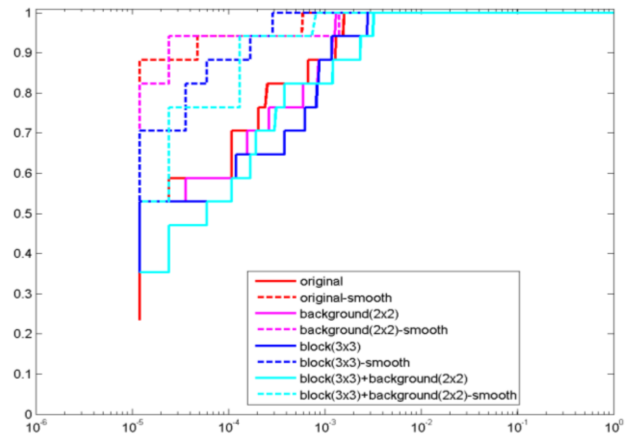
(b) Time (seconds)

Fig. 4. Comparison of RX using block (3x3) and background sub-sampling (2x2) options.

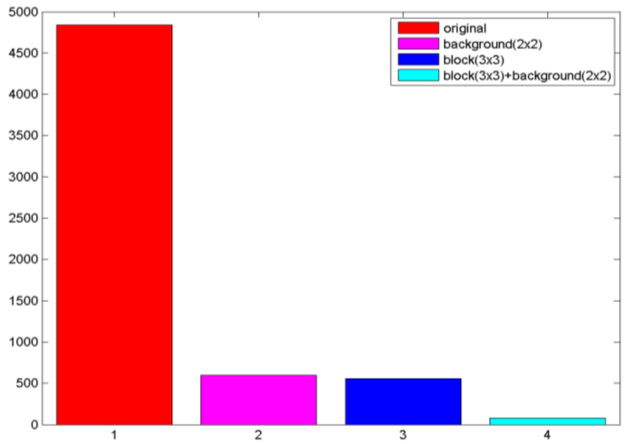
KRX results are shown in Fig. 5. We have the following observations:

- Both background sub-sampling and block pixel bring significant speed enhancement;
- Combining both options brings best speed improvement;
- Smoothing improves the anomaly detection performance significantly;
- The anomaly detection performance of using background sub-sampling is comparable to that of the original method;
- The anomaly detection result of KRX is much better than RX.

Note that the processing time of KRX with all of the options is about 78 seconds, which is 1/62 of the processing time (4,840 seconds) of KRX on the same sub-sampled (3x3) image. The processing time of KRX on the 10 band 800x1024 image was about 15 hours. So the overall reduction factor is close to 700. That is, the speed improvement is from 15 hours down to 78 seconds.



(a) ROC



(b) Time (seconds)

Fig. 5. KRX using block (3x3) and background sub-sampling (2x2) options.

VI. CONCLUSIONS

Existing anomaly detection algorithms such as RX and KRX require significant computational power in terms of many minutes/hours to process a hyperspectral image. This limits the application scope of RX and KRX in real-time applications. In this paper, we propose a systematic approach to performing fast execution of RX and KRX algorithms. First, we apply PCA to reduce the spectral dimension. Second, we apply spatial sub-sampling to reduce the spatial dimension. Third, we apply background sub-sampling and block pixel concept to further enhance the speed up. Experimental results demonstrate that the speed up factor is 220 for RX and 700 for KRX. All of experiments were done using a regu8lar PC.

Future work will include the utilization of multiple cores in PC. Nowadays, 8-core or 16-core PCs are cheaper than $1,000s. Another option is to utilize powerful GPUs.

ACKNOWLEDGMENT

This work was supported in part by the Air Force Office of Scientific Research (AFOSR) under contract FA9550-09-C-0162. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the AFOSR.

REFERENCES

[1] Hyperion, https://en.wikipedia.org/wiki/Earth_Observing-1

[2] AVIRIS, https://en.wikipedia.org/wiki/Airborne_visible/infrared_imaging_spectrometer

[3] HyspIRI, https://hyspiri.jpl.nasa.gov/

[4] GeoEye, https://en.wikipedia.org/wiki/GeoEye

[5] Landsat, http://landsat.usgs.gov/

[6] I. S. Reed and X. Yu, “Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution,” *IEEE Trans. Acoust., Speech Signal Process.*, vol. 38, no. 10, pp. 1760–1770, Oct. 1990.

[7] H. Kwon, N.M. Nasrabadi, “Kernel RX-algorithm: A nonlinear anomaly detector for hyperspectral imagery,” IEEE Transactions on Geoscience and Remote Sensing, Vol.43, No.2, Feb 2005.

[8] J. Meola, Analysis of Hyperspectral Change and Target Detection as Affected by Vegetation and Illumination Variations (Masters Thesis, University of Dayton, Dayton, OH, 2006).