Development of a library for Hyperspectral Image Analysis on the GPU platform

Students: Yamil Asusta (Undergraduate Student) and Blas Trigueros   Advisors: Nayda Santiago and Miguel Velez-Reyes
yamil.asusta@upr.edu, blas.trigueros@upr.edu   naydag.santiago@upr.edu, miguel.velez7@upr.edu

Abstract

Hyperspectral images (HSI) provide data rich information in both the spatial and spectral domain. The special properties of hyperspectral data have significantly expanded the domain of many analysis techniques, including supervised and unsupervised classification, spectral unmixing, compression, target, and anomaly detection which are of importance in hyperspectral image exploitation in defense and security applications. The main purpose of this research is to develop a robust library for hyperspectral image processing that perform efficiently on Graphics Processing Units (GPU), considerably improve the algorithm throughput, and facilitated rapid prototyping of HSI-analysis systems. This library is intended to be released as an Open Source project which will be available to the hyperspectral image processing community in different application domains. This undergraduate research project is a continuation of the work on HSI target detection on GPUs from last year.

Relevance

This work will provide a library with a set of tools to the hysperspectral image processing community to work on rapid prototyping of HSI exploitation algorithms. This will reduce time to solution and will help the HSI users community to concentrate on algorithm development and image analysis and not on implementation details.

Technical Approach

- Analyze and select parallelizable algorithms from the MATLAB Hyperspectral Image Analysis Toolbox (HIAT)[1]
- Determine detectors to be implemented[2]
- Develop library containing the selected algorithms

Previous Work

Detectors

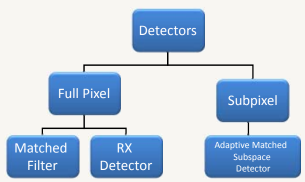

Figure 1: Implemented target detection functions

HIAT

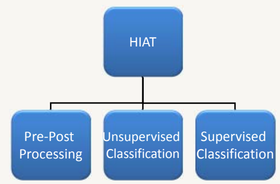

Figure 2: HIAT functions

- Matched Filter
$$y = D(x) = \frac{(\mu_1-\mu_0)^T \Gamma^{-1}(x-\mu_0)}{(\mu_1-\mu_0)^T \Gamma^{-1}(\mu_1-\mu_0)}$$
- RX Detector
$$y = D(x) = (x-\mu_0)^T \Gamma^{-1}(x-\mu_0)$$
- Adaptive Matched Subspace Detector
$$y = D(x) = \frac{x^T (P\frac{1}{B} - P\frac{1}{SB})x}{x^T P\frac{1}{SB} x}$$

Where μ_1 represents the mean of the target, μ_0 represents the mean of the background class,x represents the pixel vector, Γ represents the covariance matrix, P(1/SB) represents the orthogonal projection matrices to the background subspace, while P(1/B) represents the union of the background and target subspaces, respectively.

- Feature Extraction/Selection
  - Principal Component Analysis
  - Singular Value Decomposition
  - Discriminant Analysis
- Classifiers
  - Euclidean Distance
  - Fisher Linear Discriminant
  - Angle Detection
  - Mahalanobis Distance
  - Maximum Likelihood
- Abundance Estimators
  - Non negative sum to one
  - Non negative least square

Technical Approach

- Identify and remove proprietary code in the detectors
  - RX Detector uses Cholesky Decomposition from Intel's MKL[3]
  - $A = LL^*$
- Remove proprietary code from PCA and SVD
  - Both use SVD routine from CULA[4]
  - $M = U\Sigma V^*$
- Explore open source alternatives such as MAGMA[5]
  - Replace proprietary dependency's
- Thorough testing of all implementations
  - Benchmarking
  - Use extensive sets of Hyperspectral data
- Create documentation, tutorials, and examples

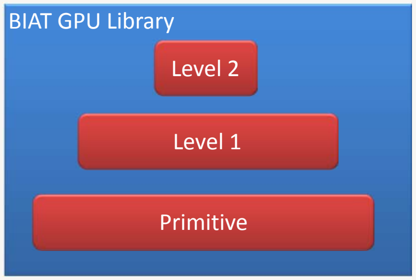

Figure 3: Functions within the library will be classified as primitive (fundamental), Level 1 (basic constructs in hyperspectral image analysis), or Level 2 (ready to use algorithms).

Accomplishments Through Current Year

- Analysis of detection algorithms and their structure.
- Proprietary code within algorithms detected in target detection code.
- Selection of detection code to be ported to GPU.
- Identified which algorithms are amenable to GPU and which are not.

Future Work

Finish Reed-Xiaoli (RX), Principal Component Analysis, and Singular Value Decomposition incorporating MAGMA. Test thoroughly and document for the library.

Opportunities for Transition to Customer

GPU implementation of detection algorithms may be incorporated in stand off and portal explosive detection systems that require high throughput.

References

1. *Rosario-Torres, Samuel; Vélez-Reyes, Miguel; Hunt, Shawn D.; and Jiménez-Rodríguez, Luis O., "The MATLAB Hyperspectral Image Analysis Toolbox" (2007). Research Thrust R3 Presentations. Paper 17.* http://hdl.handle.net/2047/d10009176
2. *Trigueros-Espinosa, Blas; Rosario-Torres, Samuel; Vélez-Reyes, Miguel; Santiago Santiago, Nayda Grisel, "GPU Implementation of Target Detection Algorithms for Explosive Material Detection using Hyperspectral Imaging and NVIDIA® GPUs"*
3. *Intel® Math Kernel Library* http://software.intel.com/en-us/articles/intel-mkl/
4. *CULA - GPU Accelerated Linear Algebra* http://www.culatools.com/
5. *MAGMA - Matrix Algebra on GPU and Multicore Architectures* http://icl.cs.utk.edu/magma/

This material is based upon work supported by the U.S. Department of Homeland Security under Award Number 2008-ST-061-ED0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied of the U.S. Department of Homeland Security.