The remote detection of explosive devices and materials has become a desirable ability and has garnered considerable interest over the last few years. Hyperspectral imagery techniques for target detection can be applied for the detection of traces of explosives, but this is not a trivial task since in most scenarios it is necessary to deal with small target signatures (explosive traces) under heavy clutter (container material). In addition, the large data volumes of hyperspectral images make this processing even more challenging, especially if the application requires the detection to be performed in real time. In this work, we propose to exploit the high parallel computation resources of GPUs for the implementation of target detection algorithms based on different background spectral variability models. The GPU implementation was done using the Compute Unified Device Architecture (CUDA) of the NVIDIA GPUs. In addition, a multi-core CPU-based implementation was developed to estimate the speedups. The detection accuracy of the implemented algorithms was evaluated using a set of phantom images simulating traces of different materials (as the explosive compound) on clothing. The final objective of this work is to study what elements in the structure of the detection algorithms make them suitable for GPU implementation, study a potential application in real-time detection of traces of explosives and develop a library of target detection algorithms to facilitate future use by researchers.

Abstract

State of the Art and Challenges

The detection of explosives can be achieved by detection of explosive vapors which escape from a container into the air [1], by detection of traces of explosive materials in clothing, containers, automobiles, etc. [1, 2], or by detection of some unusual behavior in hyperspectral imagery [3]. Hyperspectral imagery techniques have been successfully applied to gas detection [4], but the detection of vapors released by explosives is a difficult challenge since most explosives are generally formulated to have extremely low vapor pressure. The detection of explosive traces using Hyperspectral Imaging (HSI) presents issues like dealing with small target signatures under heavy clutter and the large data volumes of hyperspectral images.

Graphics Processing Units (GPUs) promise to be a suitable platform for solving hyperspectral image processing problems, because they are designed to perform intensive tasks in parallel. Tarabanja et al. [5] proposed a GPU implementation of an anomaly target detection algorithm based on normal mixture models of the background, showing that GPUs can offer enough speedup to enable real-time detection for airborne applications. In addition, recent works of A. Paz and A. Plaza [6] suggest that GPUs may offer important advantages as a computing platform in defense and security applications due to their capacity to provide high performance computing at very low costs.

Technical Approach

Four different target detection algorithms were implemented using the Compute Unified Device Architecture (CUDA) of NVIDIA: Full-pixel target detectors:

- Matched Filter Detector (MF):
  \[ y = D(x) = \frac{(\mu_i - \mu_0)}{\Gamma^{-1}(\mu_i - \mu_0)^T} x \]
  \( \mu_i \): background mean
  \( \mu_0 \): target mean
  \( \Gamma \): covariance matrix

- RX detector:
  \[ y = D(x) = x^{-1} T \Gamma^{-1} x^{-1} x \]

Subpixel target detectors:

- Adaptive Matched Subspace Detector (AMSD):
  \[ y = D(x) = x^T P_{SB} x \]
  \( P_{SB} \): orthogonal projection matrix (subspace)

- Adaptive Cosine Estimator (ACE) Detector:
  \[ y = D(x) = x^T T \Gamma x^{1/2} T \Gamma x \]

A multi-core GPU-based implementation was used as a reference to estimate the speedup of the corresponding GPU implementation. For linear algebra computations, the CUBLAS and CULA™ libraries were used in the GPU implementation. For the CPU-based implementation, the Intel® MKL library was used.

Experimental Results

The computation time of the GPU implementation was measured for the different detection algorithms and compared to the computation time of the CPU implementation running on a quad-core Intel Xeon® processor. This computation includes the memory transfer times.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>CPU (ms)</th>
<th>RX (ms)</th>
<th>ACE (ms)</th>
<th>AMSD (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MF</td>
<td>600</td>
<td>200</td>
<td>150</td>
<td>120</td>
</tr>
<tr>
<td>RX</td>
<td>120</td>
<td>50</td>
<td>40</td>
<td>30</td>
</tr>
<tr>
<td>ACE</td>
<td>120</td>
<td>40</td>
<td>30</td>
<td>20</td>
</tr>
<tr>
<td>AMSD</td>
<td>120</td>
<td>40</td>
<td>30</td>
<td>20</td>
</tr>
</tbody>
</table>

We achieved speedups between 36-70x and running times in the order of hundreds of milliseconds for the GPU implementation of the detection algorithms. In the future, we will analyze the GPU performance on larger hyperspectral images and evaluate other detection algorithms.

Research to Reality

The objective of this research is to evaluate the GPU architecture as a processing hardware platform to accelerate target detection algorithms for hyperspectral imaging. These are the first steps in the design of a standard surveillance system for real time detection of traces of explosives or hazardous materials.

Methodology

Experimental Setup:
- A scene was prepared simulating explosive traces on clothing.
- A hyperspectral image of the scene is registered using SOC-700 camera.
- The hyperspectral cube is processed using different target detection algorithms running on a NVIDIA® Tesla C1060 GPU.

Figures:
- Scene simulating traces of explosive on clothing
- Hyperspectral Camera (6.43×0.9μm)
- Hyperspectral Image (640×640 pixels, 120 bands)
- Detection Map
- Target traces
- Detection Map
- GPU-based Workstation (NVIDIA® Tesla C1060)

References