

AFRL-IF-RS-TR-2003-176
Final Technical Report
July 2003



A PROOF OF CONCEPT FOR 10x+ EFFICIENCY GAINS FOR MULTI-SENSOR DATA FUSION UTILIZING A HOWARD CASCADE PARALLEL PROCESSING SYSTEM

Booz Allen & Hamilton, Incorporated

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. N231

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

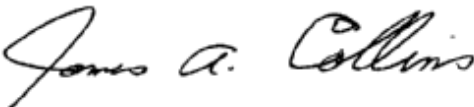
The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2003-176 has been reviewed and is approved for publication.

APPROVED: 
CHRISTOPHER J. FLYNN
Project Engineer

FOR THE DIRECTOR: 
JAMES A. COLLINS, Acting Chief
Information Technology Division
Information Directorate

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 074-0188	
Public reporting burden for this collection of information is estimated to average 1 hour 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 Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE JULY 2003	3. REPORT TYPE AND DATES COVERED Final May 02 – Nov 02	
4. TITLE AND SUBTITLE A PROOF OF CONCEPT FOR 10x+ EFFICIENCY GAINS FOR MULTI-SENSOR DATA FUSION UTILIZING A HOWARD CASCADE PARALLEL PROCESSING SYSTEM			5. FUNDING NUMBERS C - F30602-02-C-0102 PE - 61101E PR - POC1 TA - 0X WU - 01	
6. AUTHOR(S) Kevin Howard				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Prime: Booz Allen & Hamilton, Incorporated 8283 Greensboro Drive McLean Virginia 22102-3838 Sub: Massively Parallel Technology, Inc. 331 South 104 th Street, Suite C Louisville Colorado 80027			8. PERFORMING ORGANIZATION REPORT NUMBER N/A	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Defense Advanced Research Projects Agency 3701 North Fairfax Drive Arlington Virginia 22203-1714 AFRL/IFTC 26 Electronic Parkway Rome New York 13441-4514			10. SPONSORING / MONITORING AGENCY REPORT NUMBER AFRL-IF-RS-TR-2003-176	
11. SUPPLEMENTARY NOTES AFRL Project Engineer: Christopher Flynn/IFTC/(315) 330-3249/ Christopher.Flynn@rl.af.mil				
12a. DISTRIBUTION / AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.				12b. DISTRIBUTION CODE
13. ABSTRACT (Maximum 200 Words) Tools and techniques were developed by Massively Parallel Technologies Inc. (MPT) which enable intrinsically non-parallel processing problems to be processed by inexpensive parallel processing architectures. The objective of this contract effort was to create a more effective method for multi-sensor searches, and toward that goal, to improve data fusion and processing capability by >10X over current capabilities (when compared to the capability as shown in a uni-processor environment). This objective was met, and the performance speedup yielded a greater than 70X speedup on a 127-node Howard Cascade RAIS (Redundant Array of Inexpensive Systems). The processing solution implemented is original and innovative; it can be implemented in a non-intrusive manner, yielding accurate results and efficient scaling into existing processing environments, using inexpensive off-the-shelf components. In addition a methodology has been developed that provides for rapid integration of new algorithms into the defined multi-sensor fusion processing solution.				
14. SUBJECT TERMS Parallel Processing, Data Fusion			15. NUMBER OF PAGES 19	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

Table of Contents

1. Summary	1
2. Introduction.....	1
3. Methods, Assumptions and Procedures	3
4. Results and Discussion	8
5. Conclusions.....	13
6. Recommendations.....	14
7. References.....	14
8. Acronyms.....	14

List of Figures

Figure 3.1 – 8-bit grayscale “full spectrum” base image.....	4
Figure 3.2 – 8-bit grayscale “red spectrum” image	5
Figure 3.3 – 8-bit grayscale “blue spectrum” base image	6
Figure 4.1 – A red spectrum M1A1 kernel at 45 degrees orientation.....	7
Figure 4.2 – A blue spectrum M1A1 kernel at 315 degrees orientation.....	7

List of Tables

Table 5b1 Red Spectrum Results:.....	11
Table 5b2. Blue Spectrum Results:.....	12
Table 5b3 Fusion Results:.....	12
Table 5b4 Processing Speed (compared to unmodified Algorithm running on a single CPU which took 424 seconds to complete the process)	12

1. Summary

Massively Parallel Technologies (MPT, Inc.) built a process around scalable parallel processing of individual sensor data streams using a Howard Cascade© architecture machine to produce a list of items of interest within a given image. With this enhanced process, multiple sensor inputs may be fused to evaluate an area of interest and generate a list of items of interest with minimal time and manpower. To this end, MPT created the proof-of-concept demonstration detailed herein. MPT systems could easily be tactically configured to support fixed and/or transportable Intelligence/Command and Control battlefield operations.

2. Introduction

Intelligence gathering is a critical function in today's warfighting where information superiority is a force multiplier. With the advent of multiple sensor technologies and high resolution coverage, the ability of human analysts to quickly and accurately transform the collected data into usable knowledge and disseminate that information to all relevant users within tactical timelines is severely limited. Data collected, but not used effectively, is actually a force limiter because assets and forces used for the data production process detract from overall force implementation and subtract from the commander's attention to the battlefield.

As each sensor and its uses become known to enemy forces, measures may be taken to conceal items of interest or deceive the sensor. Camouflage, thermal deception, using terrain or vegetation to conceal, or even use of fake systems are examples of conceal/deceive methods. This reduces the effectiveness of the sensor and/or creates a level of uncertainty in feature identification. Feature extraction, then, requires more robust solutions using multiple sensor types or multi-date imagery, when available, to maintain required levels of certainty.

Automation of processes is necessary to fully and effectively exploit available and emerging sensor inputs. Current levels of automation do not provide the robust or timely solutions needed or are far removed from effective battlefield integration. For this study, Massively Parallel Technologies took a system-wide approach to improving the knowledge generation process that provides more robust, accurate processing while significantly decreasing production times.

The Howard Cascade Architecture is based from first principles on Amdahl's Law. Amdahl's Law, which predicts parallel performance for a given application on multiple processors.

Amdahl's law is: $\text{Speedup} = 1/((1-f) + f/p)$

Where

f	= percent of parallel activity within the algorithm
p	= number of processors
Speedup	= the processor speed multiplier 1x, 2x, etc. of the current processor speed of the individual linked processors.

Amdahl's law only takes into consideration the degree of parallel activity at the algorithm level and the number of processors used in the calculations.

Finding the limit of Amdahl's law (with respect to the number of processors) is a standard operation which yields the disheartening understanding of how little serial activity must be present before the parallel processing effect becomes unusable. This is shown below:

$$\text{Maximum Speedup} = \lim_{p \Rightarrow \infty} [1/((1-f) + f/p)] = 1/(1-f)$$

Where:

Maximum Speedup	= the processor speed multiplier 1x, 2x, etc. of the current processor speed of the individual linked processors, if there are an infinite number of processors.
f	= percent of parallel activity within the algorithm
p	= number of processors

Which yields the following table:

f	Maximum Speedup	
0.10000	1.11	Processor Equivalent
0.20000	1.25	Processor Equivalent
0.30000	1.43	Processor Equivalent
0.40000	1.67	Processor Equivalent
0.50000	2.00	Processor Equivalent
0.60000	2.50	Processor Equivalent
0.70000	3.33	Processor Equivalent
0.80000	5.00	Processor Equivalent
0.85000	6.67	Processor Equivalent
0.90000	10.00	Processor Equivalent
0.95000	20.00	Processor Equivalent
0.99000	100.00	Processor Equivalent
0.99900	1000.00	Processor Equivalent
0.99990	10000.00	Processor Equivalent
0.99999	100000.00	Processor Equivalent

Since, the key parameter in Amdahl's law is "f", Massively Parallel Technologies, Inc. has approached the problem of generating high performance with multiple processors from an algorithm-centric perspective. This is the perspective most compatible with Amdahl's law.

3. Methods, Assumptions and Procedures

1. Cluster Configuration

The demonstration trials were performed on a 127+20 node Howard Cascade[®] parallel processing system¹, where 127 nodes are used for actual data processing and 20 nodes are available as node controllers to configure and optimize the processing nodes parallelization. Each node uses an AMD Thunderbird 1330Mhz processor, running Windows NT 4.0 SP6. Inter-Cluster communication is provided by 100BaseT 3Com 3C905x NICs networked together by Hewlett-Packard 4000 Series Ethernet switches.

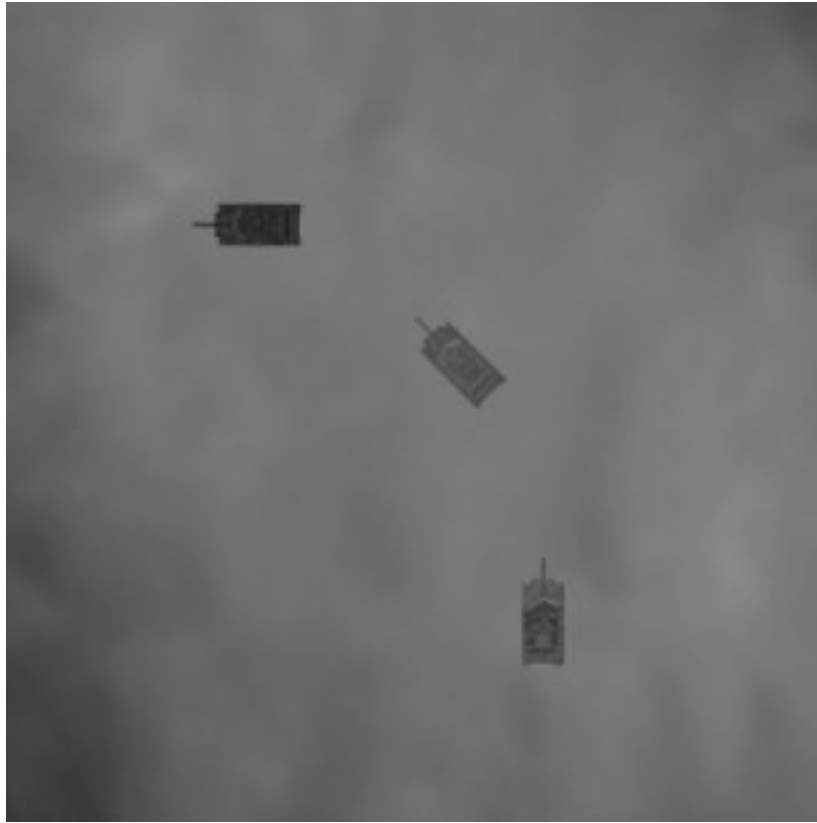
2. Client Application Configuration

The client application was written in Visual Basic utilizing DLLs written in C with Microsoft Visual Studio v.6.0. For the demonstration trials, the client application was run on a Sony VAIO laptop with Pentium III 750MHz processor. Connectivity to the cluster was achieved with a standard 100BaseT ethernet network.

3. Sensor Data

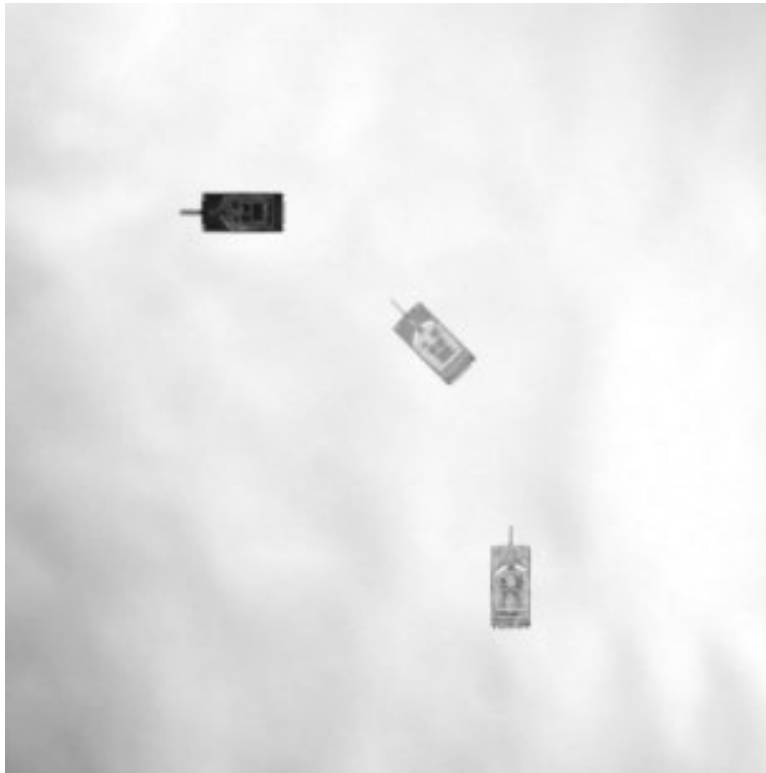
Two sensor inputs were assumed for these trials. The two sensor data sets were comprised of 8-bit images of a mock battle scene. The bandwidth range of the first sensor was in the visible-red spectrum (~700 nm) and the bandwidth range of the second sensor data set was in the visible-blue spectrum (~400 nm). Shown below is a set of images used in the trials.

Fig. 3.1 – 8-bit grayscale “full spectrum” base image



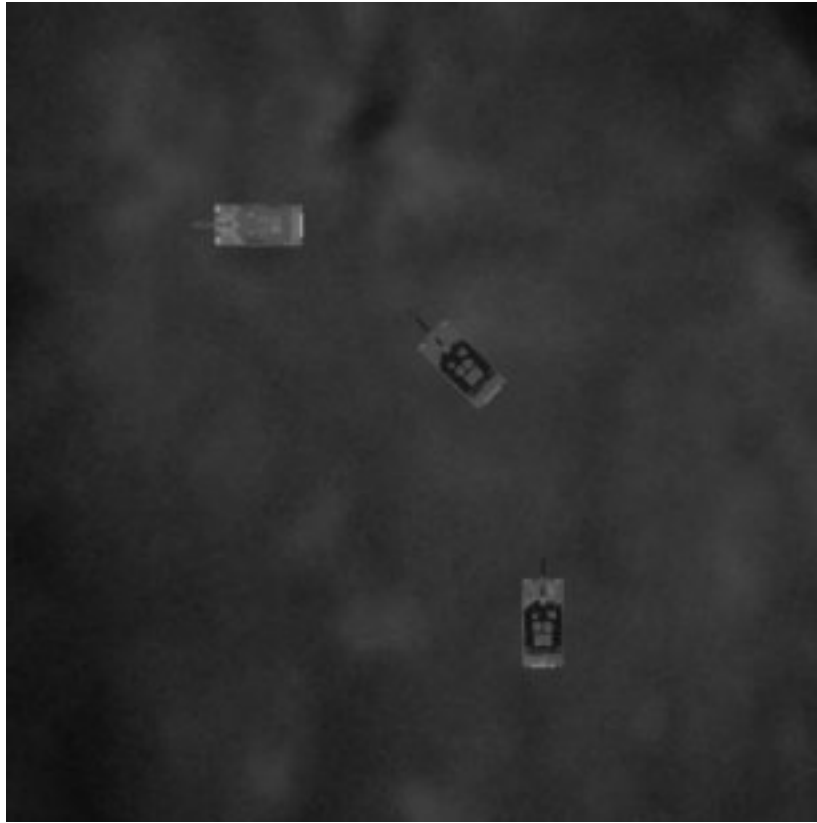
This base image shows mock M1A1 Main Battle Tanks cloaked at varying levels in the full visible spectrum.

Fig. 3.2 – 8-bit grayscale “red spectrum” image



This image shows the tanks as being ‘cloaked’ to varying degrees in the red spectrum.

Fig. 3.3 – 8-bit grayscale “blue spectrum” base image



This image shows the tanks as being ‘cloaked’ to varying degrees in the blue spectrum.

4. Database of Searchable Image Elements

Embedded in the cluster is a database comprised of known elements of interest in the various sensor regimes. These elements, known as kernels, are what will be searched for in the sensor input images. For this demonstration items of interest were constrained to individual vehicles, with future items of interest possibly including specific features of specific vehicles. Having this database available on the cluster a priori reduces the amount of communication overhead required on a particular job. The client application can request that the input images be correlated against either a subset of these kernels or the entire database. The database is designed to be dynamic; therefore, as new kernels of interest become available, they may be incorporated into the existing database.

Fig. 4.1 – A red spectrum M1A1 kernel at 45 degrees orientation

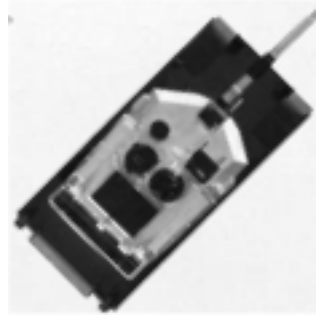
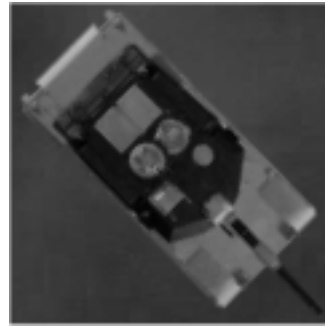


Fig 4.2 – A blue spectrum M1A1 kernel at 315 degrees orientation



For these trials shown above in figures 4.1 & 4.2, the red spectrum input images were correlated against 8 M1A1 red spectrum kernels, corresponding to 0, 45, 90, 135, 180, 225, 270, and 315 degrees of orientation. The blue spectrum input images were correlated against 8 M1A1 blue spectrum kernels, again corresponding to the same 8 orientations. The models in the battlefield scene were positioned at orientations correlating to one of the eight described rotations.

5. Correlation Algorithm

The correlation algorithm used by the cluster for these trials is the normalized cross-correlation.

Fig. 5.1 – Normalized Cross-Correlation Equation

$$C(x,y) = \frac{N \sum (P_i P_k) - \sum P_k \sum P_i}{\left[(N \sum P_i^2 - \sum P_i \sum P_i) (N \sum P_k^2 - \sum P_k \sum P_k) \right]^{\frac{1}{2}}}$$

Where: $C(x,y)$ = Correlation Coefficient at coordinate x,y
 N = Number of pixels in the kernel image
 P_k = Kernel pixel value
 P_i = Image pixel value

This algorithm was chosen because of its ability to correlate non-edge detected images, thus reducing the amount of computation required. The correlation coefficient (C) at an x,y position in the input image is always a value between -1 and 1. Values approaching 1 indicate a strong similarity between the kernel pixels and the image pixels at that x,y coordinate. The algorithm on the cluster was implemented with the constraint that given an input image and a kernel, only the maximum correlation coefficient was returned along with its x,y position in the input image.

4. Results and Discussion

a. Expert System Demonstration

1. Overview

The data flow for this demonstration was setup such that all input data was preloaded on the client application machine, and all output results were returned to the same client application machine. It should be evident, however, that from the perspective of the cluster, it is irrelevant whether the input data sets are received from a single client machine or multiple sensor platforms directly tied to the cluster.

The process began with the client application sending a job request to the cluster. This job request was for a multiple max point normalized cross-correlation operation with 8 kernels. These 8 kernels were the 8 directional kernels of the M1A1 Tank in the red spectrum. Also sent was the red spectrum input image as seen in Fig. 3.2

The cluster performed this job request, and as per the algorithm implementation, returned 8 max points corresponding to the maximum correlation coefficient for each kernel. This process was automatically repeated for the blue spectrum image and kernels.

Once both sets of max points were returned to the client application, they were then collated or ‘fused’ automatically according to a set of user-defined rules. The results were then displayed to the user as an interactive, prioritized list. Examples of this will be seen later in section 5b

2. Cluster Operation

As stated previously, the algorithm for this demonstration was configured such that a single max correlation point was returned for each kernel selected. It should be noted that an alternate configuration could have been implemented. This would be having the cluster return a set of correlation points for each kernel, with the determining factor being a correlation

threshold. The single max point configuration was selected for simplicity of demonstration of the fusion operation.

3. Client Application Operation/Sensor Fusion Overview

The client application allows the user to define a set of fusion rules a priori to be applied to the correlation results. The method of doing this involves two steps. The first is defining the fusion context, and the second involves using a simple scripting language to describe a set of prioritized rules.

4. Fusion Context

Defining the fusion context may be thought of as defining the parameters by which the script rules will apply. The main components of this are:

Fusion Trigger

This is the commonality by which the application will select max points from the individual sensor correlation lists for fusion. In this demonstration, for example, the fusion trigger was defined to be similar x,y coordinates between correlation coefficients in the separate sensor lists. So if a max point for a red spectrum kernel was near (~40 pixels) a max point in the blue spectrum, these two correlation coefficients were then fused.

Figure of Merit

The *Figure of Merit* refers to a composite value of the fused correlation coefficients. The client application allows the user to define the mathematical method of computing this. In this demonstration, a simple averaging of the two correlation coefficients was used. Other methods, such as weighted averages, or completely custom equations involving sensor probabilities may be implemented as well. Ultimately, the *Figure of Merit* should serve the purpose of giving a composite score to an interesting x,y coordinate point that will provide more insight than any single sensor result alone.

Figure of Merit Threshold

This is simply a user-defined value to which a computed *Figure of Merit* is compared. It has been set to 0.75 for this demonstration.

Sensor(n) Threshold

Again, this is a user-defined value that may be used to judge an individual sensor correlation coefficient. Both the red and blue spectrum sensors were set to a threshold of 0.65 for this demonstration. Note that the *Figure of Merit* threshold is set to 0.75. This alludes to the idea that fusing separate sensor

outputs should necessarily provide a greater degree of confidence in target identification.

Sensor(n) Probability

If the client application was integrated with a real multi-sensor platform, the probability would be a numerical description of the reliability of a given sensor of detecting a signal in its bandwidth. For this demonstration, this parameter was set to a value of 1.

5. Fusion Rules

The user of the client application may define a set of rules that will collate and prioritize the separate sensor correlation coefficients. Each defined rule will generate a list of fused correlation coefficients. These fusion lists are prioritized by the order of the defined rules. (i.e. the first defined rule assumes priority level 1).

A defined rule is one that is constructed of smaller, simpler rules known as rule primitives. These rule primitives are the atomic elements of the scripting language. Rule primitives may only return a TRUE/FALSE value. Also, a defined rule is evaluated from left to right. The currently existing rule primitives are:

$$Fm < Thr\{Fm\}, Fm > Thr\{Fm\}$$

These two rule primitives are read as “Figure of Merit Less Than Figure of Merit Threshold” and “Figure of Merit Greater Than Figure of Merit Threshold”, respectively.

$$S1 < Thr\{S1\}, S1 > Thr\{S1\}$$

These two rule primitives are read as “Sensor 1 Correlation Coefficient Less Than Sensor 1 Threshold” and “Sensor 1 Correlation Coefficient Greater Than Sensor 1 Threshold”, respectively.

$$S2 < Thr\{S2\}, S2 > Thr\{S2\}$$

These two rule primitives are read as “Sensor 2 Correlation Coefficient Less Than Sensor 2 Threshold” and “Sensor 2 Correlation Coefficient Greater Than Sensor 2 Threshold”, respectively.

For this demonstration, the following rules were defined:

Priority Level 1 Rule Description:

“If the Figure of Merit is greater than the Figure of Merit Threshold” – What this equates to is that any fused red and blue coordinate point that exceeds the

Figure of Merit threshold should be classified as a priority level 1 point of interest.

Priority Level 1 Rule Script:

$$Fm > Thr\{Fm\}$$

Priority Level 2 Rule Description:

“If Sensor 1 is greater than the Sensor 1 Threshold, OR Sensor 2 is greater than the Sensor 2 threshold, AND the Figure of Merit is less than the Figure of Merit Threshold” – This states that if either the red or blue coordinate point exceeds its threshold, but the Figure of Merit is below the Figure of Merit Threshold, this is a priority level 2 point of interest.

Priority Level 2 Rule Script:

$$S1 > Thr\{S1\} \text{ OR } S2 > Thr\{S2\} \text{ AND } Fm < Thr\{Fm\}$$

Priority Level 3 Rule Description:

“If Sensor 1 is less than the Sensor 1 Threshold, AND Sensor 2 is less than the Sensor 2 threshold, AND the Figure of Merit is less than the Figure of Merit Threshold”- This states that even though both sensors are below their respective thresholds, and the Figure of Merit is below its threshold, because both red and blue coordinate points are similar, it is to be labeled as a priority level 3 point of interest.

Priority Level 3 Rule Script:

$$S1 < Thr\{S1\} \text{ AND } S2 < Thr\{S2\} \text{ AND } Fm < Thr\{Fm\}$$

b. Sensor Fusion Informational Results

This study demonstrated the value of fusing the correlation operations from multiple sensor inputs. It not only reduces the overall effectiveness of cloaking in any one particular sensor regime, but also increases the probability of detection when an item of interest is cloaked in all sensor regimes by corroborating those detections that may fall below detection thresholds.

Table 5b1 Red Spectrum Results:

Orientation:	Correlation Coefficient:
0	.78
45	.46

90	.86
135	.99
180	.89
225	.44
270	.70
315	.84

Table 5b2. Blue Spectrum Results:

Orientation:	Correlation Coefficient:
0	.43
45	.32
90	.95
135	.74
180	.71
225	.33
270	.44
315	.38

Table 5b3 Fusion Results:

Orientation:	Figure of Merit:	Priority:
90	.91	1
135	.87	1
0	.61	2
315	.59	2
270	.57	3

Table 5b4 Processing Speed (compared to unmodified Algorithm running on a single CPU which took 424 seconds to complete the process)

Number Of Nodes	Time in Seconds	Speedup %	Per Processor Parallel Efficiency	Algorithmic Parallelization Efficiency
1	427	0.99	.99	not applicable
3	21	3.53	1.16	1.07494
7	56	7.63	1.08	1.01366
15	29	14.72	.97	0.99866
31	15	28.47	.91	0.99703
63	9	47.44	.75	0.99471
127	6	71.17	.56	0.99377

The Per Processor Parallel efficiency reflects the percentage of the processor capability used to process its parallelized portion of the algorithm. As expected with efficient parallelization, each part of the algorithm to be processed became smaller as it was divided up into more processors. Consequently less and less processor power and overhead was required from each processor. One might note that running the original algorithm on a single processor (not on the interconnected cluster of CPUs) is about 1% more efficient than a single processor running the parallelized form on the cluster (I.E. not designed for single processor improvement, performance improvement comes with multiple processors).

Classic attempts at parallelizing this type of problem generally run into efficiency problems where interprocessor communication, memory management, and other issues result in creating more overhead impacts and latencies which eclipse the potential improvement possible having additional processors available. The measured results using the Howard Cascade and MPT tools and techniques show that efficiency is maintained as more processors are used. This Algorithmic Parallelization efficiency, f , reflects the percentage of the algorithm done in parallel by the multiple processors. A value of one reflects that all of the speed up processing is done through the parallelized processors. The cases using 3 & 7 processors having efficiencies greater than one were associated with added memory and communication efficiencies afforded by the algorithm and its cluster architecture.

5. Conclusions

The objective of this contract effort was to create a more effective method for multi-sensor searches, and toward that goal, to improve data fusion and processing capability by >10X over current capabilities (when compared to the capability as shown in a uni-processor environment). This objective has been met, and the performance speedup yielded a greater than 70X speedup on a 127-node Howard Cascade RAIS (Redundant Array of Inexpensive Systems). The processing solution implemented is original and innovative; it can be implemented in a non-intrusive manner, yielding accurate results and efficient scaling into existing processing environments, using inexpensive off-the-shelf components. In developing and testing our solution for multi-sensor data fusion, a calibrated imagery baseline has been established that may be extrapolated to multi-sensor mission scenarios: these first iteration database elements were generated from a controlled environment, and subsequent database elements should be from operational satellite imagery to validate the efficacy of the application of this technology to actual current need. Lastly, a methodology has been developed that provides for rapid integration of new algorithms into the defined multi-sensor fusion processing solution.

6. Recommendations

Global awareness and command and control are required to give the warfighter leverage and competitive advantages. Global Engagement addresses the need for the capability to dominate an opponent across the range of military operations -- full spectrum dominance. Full spectrum dominance requires information superiority: the ability to collect, process, analyze and disseminate information.

The subject demonstration outlined a capability with many flexible application possibilities for achieving the objective of full spectrum dominance. MPT systems could be tactically configured to support fixed and/or transportable Intelligence/Command and Control battlefield operations in enhancing global awareness.

Further study with the objective to refine the parallel multi-sensor fusion methods defined here would yield vast improvements in operational capabilities in the following areas:

- ❑ Target Recognition and Database Assessment
 - Additional spectral bands
 - Additional raw satellite imagery
 - Full slant-angle
 - Arbitrary rotation angle
 - Obscured/Partial target visibility
 - 2D to 3D Morphology
- ❑ Moving Target Indication
- ❑ Change Detection and Tracking

7. References

¹ For more information on the Howard Cascade architecture, please refer to the white paper “**An Evaluation of the Howard Cascade Parallel Processing Method Using Amdahl's Law**” by Kevin Howard, CTO, Massively Parallel Technologies

8. Acronyms

2D	Two Dimensions
3D	Three Dimensions
CPU	Central Processing Unit
DLL	Dynamic Linked Libraries
DoD	Department of Defense
MPT	Massively Parallel Technologies Inc. http://www.massivelyparallel.com/
nm	Nanometers
PC	Personal Computer