Virtual Hammer

Paul Metzger, Lawrence Bush, Peter Mastromarino, Steven Rak, and Tod Shannon

Rapid development of new techniques and technologies for integrated sensing and decision support (ISDS) requires a streamlined and efficient way to conduct research. With a simulation testbed emulating the intelligence, surveillance, and reconnaissance (ISR) sensors present during an exercise like the Silent Hammer experiment, and tools to measure the human decision processes taking place, we could make significant progress in the advancement of ISDS technologies. An interactive, operator-in-the-loop simulation environment such as this—Virtual Hammer—has been developed and is currently being used to explore the potential for synoptic-level sensors to cue unmanned aircraft.

In October 2004, a Navy Sea Trial Limited Objective Experiment named Silent Hammer took place on and around San Clemente Island, off the southern coast of California. The goal of the experiment was to exercise advanced technologies in a joint Global War on Terror operation consisting of Special Operations Forces and U.S. Marines based on an SSGN (a U.S. Navy nuclear-powered cruise missile submarine), aided by application of advanced unmanned systems. The experiment was highly successful. Although the execution phase of Silent Hammer was only ten days, however, the complexity of such a large-scale field experiment required 62 organizations and 1135 personnel a daunting 18 months to plan.

Several technical advancements in the integrated sensing and decision support (ISDS) domain were implemented and demonstrated during Silent Hammer (as outlined elsewhere in this issue of the *Lincoln Laboratory Journal*). In order for ISDS to achieve more rapid progress on the development of new techniques and technologies, however, a more streamlined and efficient way of conducting this research was needed.

The solution was interactive, operator-in-the-loop (OITL) simulation. Given a simulation testbed that could emulate the intelligence, surveillance, and reconnaissance (ISR) sensors present during a large scale field exercise like Silent Hammer, and provided with enough tools to record and measure the human decision processes that take place during these experiments, one could make significant progress in the advancement of ISDS technologies without the need to conduct all of the research during a field exercise. The Virtual Hammer project illustrates the use of simulation and modeling to extend the field exercise experience by introducing sensor information architecture excursions for comparative performance assessment.

We need to first make two important points regarding the use of simulation as a medium for ISDS research. First, the simulations need to have adequate fidelity to provide a high degree of confidence that the results obtained in the simulation environment will faithfully reflect what would happen on the ground. Second, we do not suggest that simulations replace the need for field experiments. The use of both environments is key to accelerated development cycles and the rapid transition of high-value technologies. Simulation enables us to explore initial concepts and scenarios rapidly and cost effectively via simulation; these concepts can then be proven with field experiments, which have much stronger correlation to real-world operations than any simulation ever could. Moreover, data collected during field experiments can refine the simulation models and improve their fidelity.

The first experiment to be conducted in the Virtual Hammer environment explores the potential for a synoptic-level radar sensor to cue an electro-optical sensor for identification. More specifically, it seeks to answer the question of whether surface moving target indicator

FIGURE 1. The synthetic San Clemente Island (right) viewed from a similar perspective as the real-world photograph of the same locality (left).

(SMTI) radar data can be used to cue a Predator unmanned aerial vehicle (UAV) to more effectively find and identify convoys. This article describes the simulation environment that has been developed, provides an overview of the sensor modalities now implemented, and illustrates the data collection systems used during our experiments. We also discuss our SMTI exploitation algorithms, along with preliminary experimentation results. We conclude with a look at future plans and excursions beyond the Silent Hammer environment.

Overview of Virtual Hammer

The ability to perform continuous experiments in a simulation environment provides the opportunity to "dry-run" field exercises, to experiment with sensor tasking concepts of operation (CONOPS), to explore various data link and bandwidth limitations, and to support the development of data exploitation algorithms. It also provides an easily reconfigurable framework within which to model numerous ground force objective scenarios and mission threads.

The starting point for these types of virtual environments is the representation of the physical world—the digital dirt. Our participation in Silent Hammer provided us with a wealth of field data that we were able to use for comparison of our simulation models to the real world. One of the most time-consuming aspects of developing these environments is the creation of the underlying 3-D visual system database. We were fortunate enough to leverage the work of Naval Air Training Systems Division, who had, in the course of a previous project, already sponsored the development of a synthetic recreation of San Clemente Island, as shown in Figure 1.

To ensure interoperability among the suite of simulation tools required the existence of a common terrain database representation. The original database was provided in OpenFlight format [1], and the suite of tools that were selected for the Virtual Hammer environment were required to support import of this OpenFlight database.

Real-Time Simulation

Before we present details of the architecture, we should clarify exactly what we mean by the term simulation. The task at hand involve a two-dimensional construct. The horizontal dimension represents the progression of time, with simulations broken down into one of two categories: real time and non-real time. The vertical dimension represents interactivity, again broken down into two broad categories: operator in the loop (in which a human is required to interact with the simulation) and constructive (wherein a representative model of human or system behavior is used). Since part of our objectives was to perform CONOPS experimentation on concepts of operation and analysis of decision support exploitation, we decided to develop a real-time, operator-in-the-loop interactive simulation environment.

System Architecture

Figure 2 shows the Virtual Hammer system architecture. Various distributed interactive simulation (DIS) tools are shown connected by a thick black line. The surveillance products such as video, synthetic aperture radar (SAR), and SMTI pass through bandwidth emulators prior to dissemination via a metadata server utilizing a publish/subscribe dissemination scheme. The desired outputs from the Virtual Hammer simulation are ISR data products—video and still images from UAVs (both file-based and streaming video), SAR images, and SMTI detection records. The UAV video format is MPEG2 [2], supporting telemetry data as an embedded Key/Length/Value metadata stream. The SAR images are encoded as georeferenced National Imagery Transmission Format (NITF) version 2.1 files [3]. The SMTI collection records are broadcast as User Datagram Protocol/Internet Protocol (UDP/IP) network data in Stanag 4607 format [4].

One of our main goals was to create the ISR data products in the exact same formats used by real-world systems. This formatting would provide a strong framework for the development of sidecars for field experimentation. A sidecar is a set of test equipment and software that is brought to a field experiment and that operates alongside the real-world fielded systems. It accesses the same data, without adversely effecting the deployed systems. A sidecar is sometimes referred to as being "online" but not "in line" with the information flow. For development of these sidecars, in the laboratory we would stimulate them with simulated ISR data, quickly resolve any problems at the testbed, then bring the sidecar out to a field experiment to operate on real-world data of the same format as was used in the laboratory.

To recreate the airborne sensor systems, we had to disassociate simulation of the airframe from simulation of the sensor. This modularity provides us the ability to replay the same flight-path data in various experiments, but explore the use of different sensor payloads. We used aircraft navigation data collected during Silent Hammer to set the position and orientation of a simulated entity representing the airframe. For the UAV sensor, a network-capable visual system was then logically attached to the entity state data on the network, effectively slaving the attachment point of the sensor to a location on

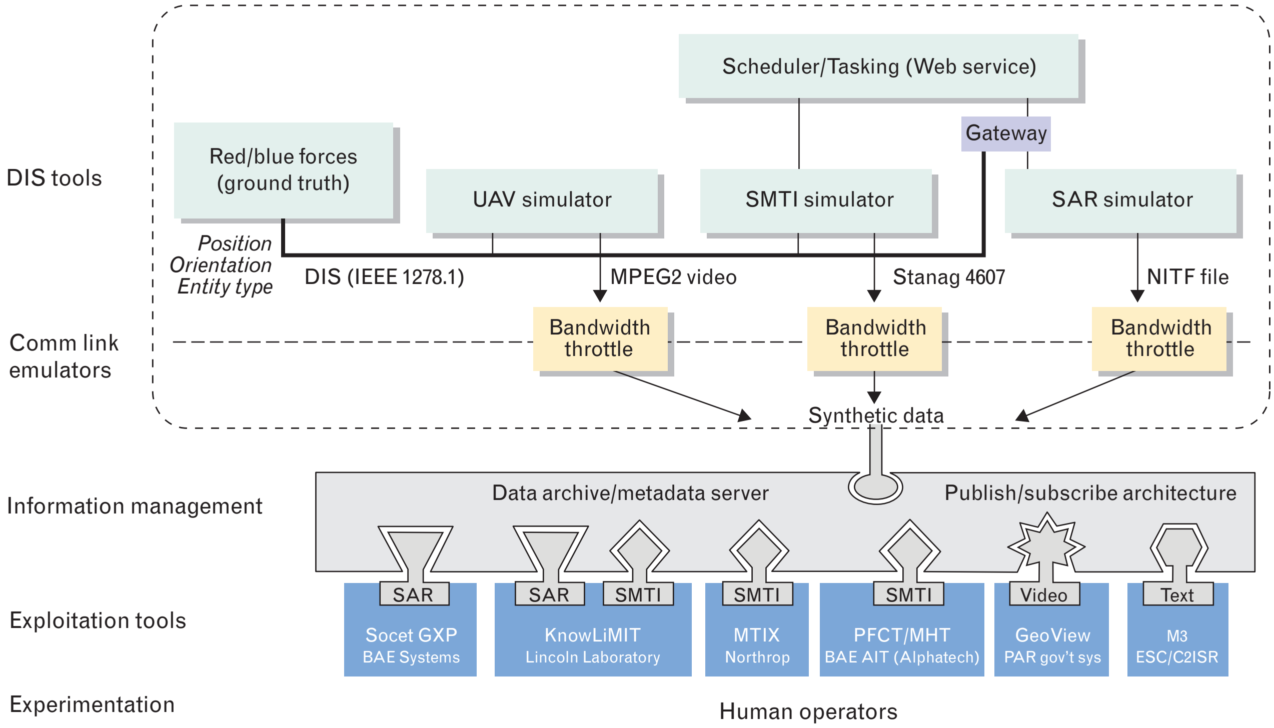


FIGURE 2. The Virtual Hammer system architecture. Simulators modeling intelligence, surveillance, and reconnaissance (ISR) interact with a representation of ground truth via the distributed interactive simulation (DIS) network. These models generate ISR data products that are then archived in a metadata server for dissemination to exploitation systems.

the airframe. Similarly, the collection of radar data was performed by associating the sensor software model with the location of an aircraft being simulated on the network. This modular separation of sensor payload from physical platform also provides the foundation for future work on the Space Radar program; we will be able to easily replace the flight path of the Boeing 707 with the orbital path of a satellite, while keeping the same sensor payload and simulation software.

One of the first hurdles we ran into on the Virtual Hammer project was finding a tool that supported the specification of spatio-temporal waypoints, performed position and orientation interpolation between these waypoints, and transmitted the results onto the simulation network. We found no suitable off-the-shelf tool, and so wrote a platform simulator to support this capability—to play back ground-truth data recorded via Global Positioning System (GPS) transponders (in the case of instrumented ground vehicles) and GPS plus inertial navigation system data, in the case of the airborne platforms.

Distributed Interactive Simulation

The ability to link heterogeneous simulations via local and wide area networks has been around since the 1980s, through the development of systems such as SIMNET [5] and NPSNET [6]. These systems, although fielded and operational, were large, stove-piped systems. During the early 1990s, the SIMNET data distribution protocols underwent a standardization effort, resulting in the creation of the Distributed Interactive Simulation, or DIS, network protocols. The DIS protocols eventually went on to become an IEEE standard (IEEE 1278.1) [7].

Although both SIMNET and DIS protocols were widely implemented, they suffered from a major drawback: the packet definitions were very rigid, making it difficult to add new types of networked functionality to the simulation environment. This constraint, in conjunction with a desire to have packet distribution schemes other than UDP/IP broadcast (as was specified by the DIS standard), led to the next evolution in simulation network protocols, the High Level Architecture (HLA) [8]. The main advantage of HLA is that it brings the concepts of object-oriented programming to network simulation protocols. Its downside is that for exercise participants to agree on the objects to be shared requires a significant amount of up-front coordination.

In addition to simulations of fictitious scenarios and vignettes, recreations of real-world events such as the Gulf War I Battle of 73 Easting [9] and the victory at Mazar-e Sharif [10] in Afghanistan have also been developed. What differentiates the Virtual Hammer environment from these other simulations is the emphasis on simulation of the pre-engagement phase of conflict—the ISR phase—plus a focus on combined air and space assets, the simulation of radar systems to generate SAR and SMTI products, and the basing of these capabilities on real-world data collections.

Ground-Truth Generation

Several different solutions were combined to represent ground-truth states, all of which transmitted DIS entity state data. A platform simulator was developed to play back recorded navigation data as well as GPS of the instrumented vehicles, with civilian vehicle traffic on the island modeled through Toyon's Ground Vehicle Simulator (GVS) [11]. The fidelity of the GVS ground vehicle positions were a bit coarser than was desired for our electro-optical (EO) simulators. This coarseness resulted in numerous database correlation issues, in which the vehicles would appear to float above the ground or burrow into the terrain. The height anomalies were fixed by using a graphics rendering feature that clamped the vehicles to the surface of the virtual terrain. GVS also had a slightly different road network, requiring the vertices of the visual-database road network of San Clemente Island to be manually entered into the simulator in order to get the two different road networks to correlate.

Tactically relevant scenarios (such as convoys) were generated with GVS when scripted behavior was adequate. Higher-level autonomous control was obtained by using a semi-automated forces (SAF) simulator. Since our scenarios took place in both ground and littoral regions, the SAF system of choice for us was VR-Forces from MäK Technologies [12]. The human animation component (for simulation of special operations forces) was provided through the use of VR-Forces and the DI Guy animation package from Boston Dynamics [13].

Simulating the Lincoln Laboratory Fleet

With the synthetic environment and simulated ground truth in place, the final technical piece of Virtual Hammer was development of a mechanism to task the sensors and collect simulated ISR data.

One of the main sensors at Silent Hammer was the

Lincoln Multi-Mission ISR Testbed (LiMIT) radar [14]. LiMIT is tasked in a round-robin fashion from a set of operator-provided aim points; updates made by the operator on board the 707 during the flight specify which aim-point group to collect next. A post-flight spreadsheet is then generated that enumerates the results of this tasking as a list of discrete SMTI and SAR collections. Virtual Hammer ingests the resulting LiMIT spreadsheets, which contain timestamps, aim-point locations, and sensor modes (SAR versus SMTI). The platform simulator maintains simulation time, and at the appropriate times sends collection request messages to either the SMTI radar model or to a software gateway used for the generation of a SAR image. UAV tasking is a two-part process. The flight path is defined pre-mission, with playback of the recorded navigation data adequate to recreate the flight. The orientation of the UAV camera is not automated: the operator is directed where to look through a combination of verbal commands and text chat. Virtual Hammer provides an identical mechanism as the baseline for UAV operation.

Electro-Optical Sensors

The tool that was chosen to emulate the UAV visual system is the Virtual Reality Scene Generator (VRSG) product from MetaVR [15]. Figure 3 shows an image generated by VRSG compared to a real-world aerial photograph.

A DISTRIBUTED SIMULATION PRIMER

The underlying premise of distributed interactive simulation (DIS) is the use of "selective fidelity" for all players in a distributed, networked virtual environment. The truth state of the world is shared by all participants. It is up to the receiver of the data, however, to decide the fidelity with which to display the data—perhaps as a 2-D map, perhaps as an interactive 3-D display, or perhaps as a radar image.

Each DIS network packet—called a Protocol Data Unit (PDU)—has a header that defines what version of DIS is running and what exercise identifier the packet is part of. This configuration allows multiple, simultaneous simulations to take place on the same network without interfering with one another. Two other fields in the header identify the protocol family and PDU kind; together, this information uniquely identifies the remainder of the packet data, and allows the receiver to decode the data.

The most common packet on the network is the Entity State PDU, which transmits the position, orientation, and velocity of all vehicles on the network. In addition to this state data, a set of seven enumerated data fields define what the vehicle is, such as a US M1A1 Abrams tank, a Soviet Mi24 Hind helicopter, or a wheeled civilian truck. Locations are transmitted in geocentric coordinates, a Cartesian system with origin at the center of the earth, $+x$ passing through the prime meridian at the equator and $+z$ through the North Pole.

To minimize bandwidth consumption, DIS implements a set of extrapolation business rules between sender and receiver called "dead reckoning." The sender runs both a high-fidelity model of his position and orientation and a low-fidelity extrapolation model. When the differences between the two exceed a threshold (in terms of position or orientation) a new Entity State PDU is sent.

On the receiving side, when an Entity State PDU is received, the extrapolation model is started and continues until either new state data arrives or a timeout occurs (generally, after 15 seconds) in which case the entity is assumed to no longer exist. With the use of dead reckoning, high-fidelity simulations running at a 30 to 60 Hz frame rate typically won't exceed a transmit rate of more than one to three packets per second.

Protocol Data Units also exist that define Fire and Detonation packets (for direct and indirect fire simulation), Collisions, Emissions and Signals, and Repair and Resupply, as well as a variety of packet types for managing the starting, pausing, and stopping of exercises.

FIGURE 3. Real-world (left) and virtual (right) views of San Clemente Island. The real-world image is a still frame from video taken by the Pelican aircraft (a Predator surrogate) flown during Silent Hammer. The Pelican is a piloted, single-engine aircraft developed and operated by the Naval Postgraduate School. The virtual image is from the Virtual Reality Scene Generator (VRSG), drawing on visual information from the synthetic San Clemente Island database.

VRSG runs on a standard Windows PC, and uses the Microsoft DirectX graphics subsystem as the core of its rendering engine. We selected VRSG over other PC-based rendering systems for three reasons. First, VRSG comes with a variety of UAV heads up display 2-D overlays already implemented (including one for Predator). Second, in addition to being a stand-alone program, VRSG has an application programmer's interface that allows us to implement features beyond the standard capabilities. Third, VRSG is rapidly becoming the de facto standard for visual simulation in the military training community, and the graphics performance and visual quality exceed that of other commercial vendors.

Surface Moving Target Indicator Radar

Our top priority in an SMTI simulation system was speed—we needed the SMTI model to execute in or near real time. Fidelity was less important. We explored several SMTI simulations, but each of them had at least one significant shortcoming. Either they were stand-alone systems that didn't integrate well in a larger system, they simulated a small focused region of interest instead of global coverage, or they had tasking and scheduler interfaces that were too rigid and inflexible.

We ultimately decided to develop a medium-fidelity model of our own to provide the SMTI simulation. Our real-time, Java-based SMTI simulation operates by first sampling the DIS network for all nonstationary entities within the approximate beam footprint of the radar (using a trapezoid to represent the area of the beam dwell). The model takes into account only moving entities; stationary vehicles will fall far below any radar's threshold of minimal detectable velocity. We chose a signal-to-noise ratio (SNR) threshold value based on the desired probability of false alarm (P_{fa}) [16]. The radar parameters of LiMIT (at the 180 MHz mode) and the associated parameters of our SMTI simulation model are shown in Table 1. This simulation model represents a probability of detection (P_d) of 95%.

The next step in our simulation of SMTI detections is to calculate the radar cross section (RCS) of each DIS entity in the dwell. RCS represents the magnitude of the echo signal returned to the radar by the target. (Since our simulation doesn't model radiated energy, we use a Swerling 2 model [17] to generate the RCS values by using look-up tables.) Using the RCS value and the radar equation [18], we then calculate SNR. If the SNR is above our threshold, the entity becomes a detection. The detection coordinate is then displaced in cross-range by using a Gaussian distribution, with the cross-range error (1/SNR) as the standard deviation. The detections are packed in a Stanag 4607 packet and transmitted onto the network.

Our model assumes that the only source of false alarms is receiver noise. LiMIT, being a real system, also generates false alarms from clutter discretes. In an attempt to match LiMIT's false-alarm rate, we lowered our SNR threshold. The SNR threshold set in our simulation represents a P_{fa} of 1.0×10^{-6} (0.000001 false alarms per beam dwell). Experimentation has shown that the SNR values generated for simulated false alarms are realistic enough to allow false-alarm mitigation algorithms to work.

Synthetic Aperture Radar

We found a few commercial SAR data simulation products that had the potential for meeting our objectives of near-real-time data production, offscreen rendering to generate large-format images, and export of images in geolocated NITF files. The synthetic SAR images required for Virtual Hammer do not need to be of the same quality as those required to train an image analyst [19]. Even the cruder simulation images, however, provide an extremely useful stimulus for a large variety of experiments.

We chose a product called the Radar Tool Kit (RTK) from Camber Corp. [20]. A comparison of real-world SAR imagery collected during Silent Hammer and simulated imagery of the same location is shown in Figure 4. RTK is physics-based radar software implementing ground-mapping Digital Radar Land Mass Simulation models, terrain and feature elevation, feature type and surface materials, and a large number of moving models (targets). Since the RTK does not directly listen to a simulation network, a gateway needed to be developed. RTK has a network socket host interface, so the primary role of the gateway is to translate DIS state data into RTK host messages. RTK uses an OpenFlight database as a starting point, but requires an offline preprocessing phase to convert the visual system texture maps into more representative radar data. Figure 5 shows one of the simulated SAR images of San Clemente Island geolocated on a map display.

Table 1. LiMIT versus SMTI Simulation

Parameter	*LiMIT (180 MHz mode)*	*SMTI Model*
Footprint range	4 km	4 km
Footprint cross section	1.5 km	1.5 km
Transmit gain	31 dB	31 dB
Effective area of antenna	–10 dB	–10 dB
Wavelength	3 cm	3 cm
Antenna length	48 cm	48 cm
Antenna height	18 cm	18 cm
Number of antenna elements	32	32
Average power	100 mW	100 mW
Time on target	60 msec	60 msec
Noise temperature	290 K	290 K
Radar losses	9 dB	9 dB
Platform velocity	180 m/sec	180 m/sec
SNR detection threshold	15 dB	1.4 dB
Threshold includes clutter?	yes	no
False alarms from noise?	yes	yes
False alarms from clutter?	yes	no

Ladar

Three-dimensional laser radar (ladar) systems can be powerful components of the Virtual Hammer simulation system. Ladar can be used as a source of simulated ground truth as well as an output product from the simulation. For many simulation applications it is necessary to have a shared high-fidelity representation of the environment across all the systems taking part in the simulation.

A real-world ladar system can rapidly map a large complex environment, such as a major city, in a period of hours. The resulting 3-D data can be used to generate the geometry of the ground environment as well as vegetation and human-built structures. In the case of San Clemente Island, a large high-quality database was already available. In future rapid-response scenarios, however, it may be necessary to construct a new database of some hotspot for use in planning and simulation. Ladar mapping can provide at least part of the rapid high-quality data required for such a database.

Work is in progress to integrate ladar data into the visual database format used by Virtual Hammer. Future work will allow the overlay of video on the ladar data, resulting in highly accurate geolocated video mapped to a 3-D surface and displayed in real time [21].

The other dimension of ladar integration is the generation of simulated ladar imagery from the virtual envi-

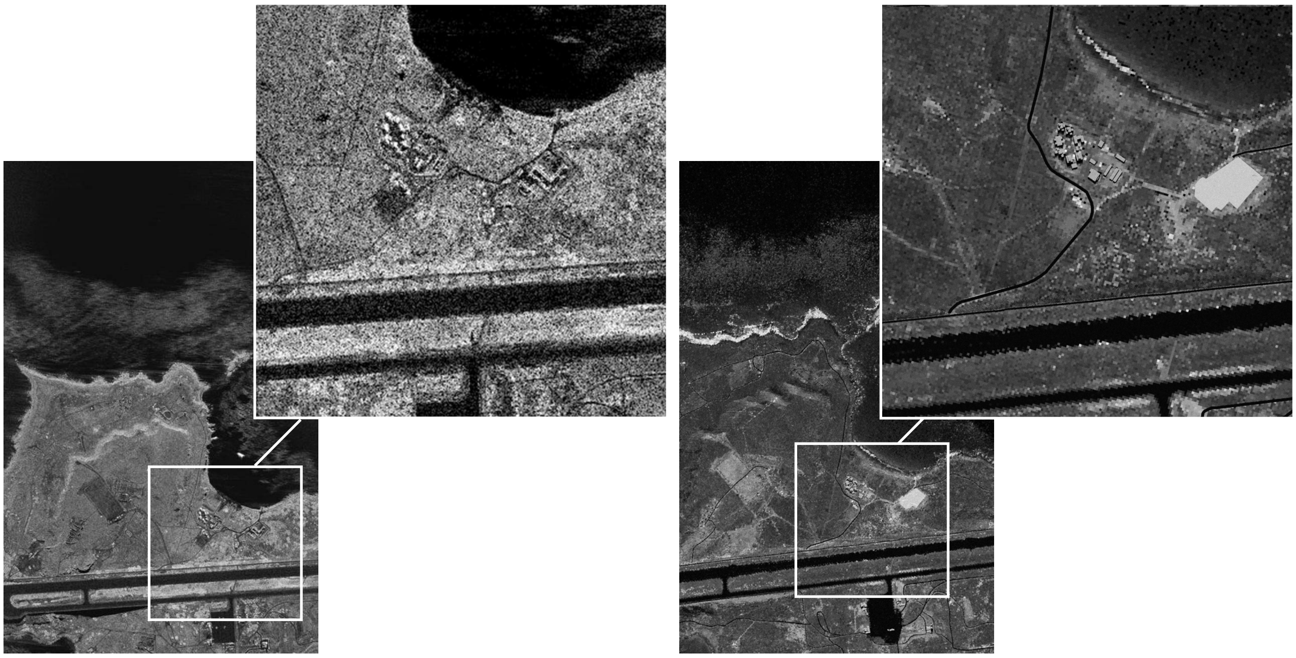


FIGURE 4. A real-world synthetic-aperture-radar (SAR) image (left) and zoomed inset of a small compound, with a simulated SAR image (right) and zoomed inset of the same compound. Both perspectives are from the northern tip of San Clemente Island, just north of the airfield, and include a portion of Northwest Harbor.

ronment. Ladar imagery could be generated either from coarse real-time models or high-fidelity engineering models, depending on the requirements of an experiment. Use of simulated ladar data would allow experiments evaluating innovative visualizations and analysis in regions of the world where no real ladar data have yet been collected. Since tools for the warfighter to utilize 3-D ladar are a developing technology, integrated, operator-in-the-loop simulation is an excellent environment to develop and test innovative uses of the data.

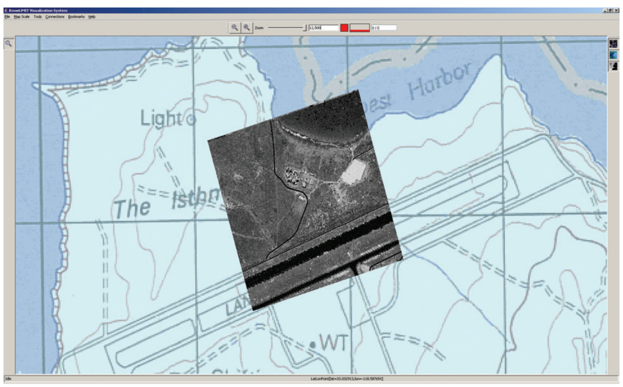


FIGURE 5. The simulated SAR zoom inset from Figure 4 is show in its geolocated position over a map of San Clemente Island. Simulated SAR imagery is generated in National Imagery Transmission Format, and includes the necessary georeference metadata. The display tool is KnowLiMIT, a program developed at Lincoln Laboratory for SAR imagery and surface moving target indicator (SMTI) radar data.

Operator-in-the-Loop Experiments

We now turn our attention from the simulation infrastructure to the most important element of the ISR processing chain—the human operator. The prime focus of the ISDS laboratory is to monitor and evaluate operators performing typical ISR tasks with the goal to identify solutions that enhance or optimize their mission performance. Thus we need a way to relate measures of performance, such as time spent searching, to measures of effectiveness, such as targets found.

The impetus for assembling the breadth and depth of the Virtual Hammer simulation environment is to provide a sufficiently rich environment that operators faithfully and rigorously perform their ISR tasks. The quality of observations, analyses, and conclusions drawn from measurements taken in this environment rests squarely on the operator's becoming intellectually and emotionally invested in the experiment at hand. That said, the environment really only has to be good enough for the operator to remain comfortable and partici-

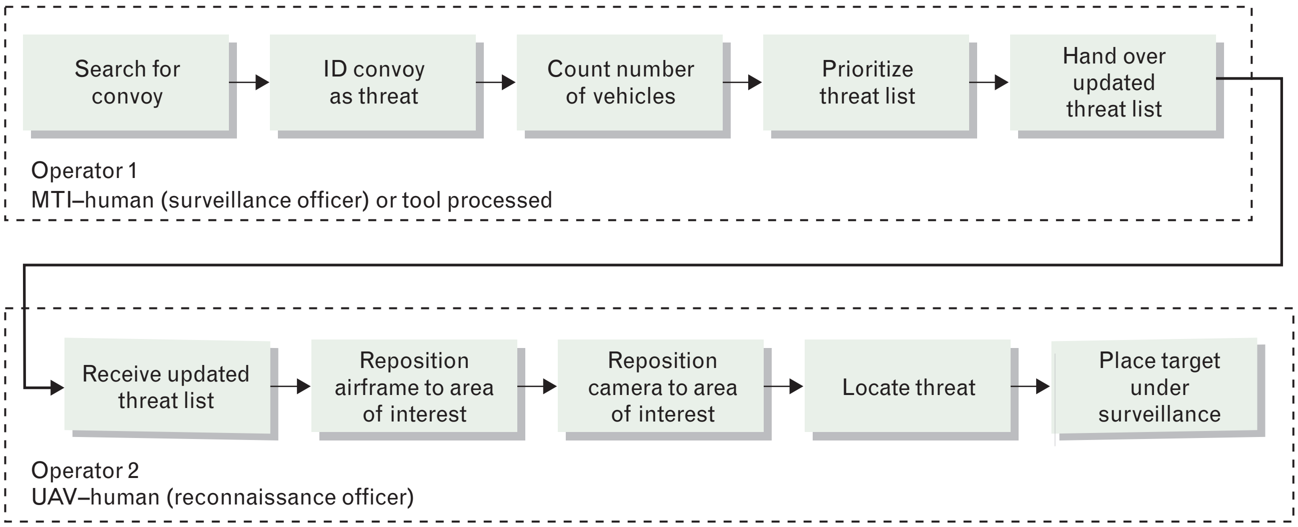


FIGURE 6. This diagram shows the sequence of ISR tasks as the search for targets transitions from a surveillance officer to a reconnaissance officer. In the initial Virtual Hammer experiments, the emphasis has been placed on modeling surveillance officer tasks up to the generation of the prioritized threat list.

pate meaningfully in the experiment. The measures of performance we collect must be normalized, as best as possible, against any low-fidelity or non-realistic representations of the virtual world in which the operator is processing.

Figure 6 provides a task breakdown for two operators participating in a target detection mission. An operator searching for a convoy in SMTI data goes through the initial steps before handing it off to the operator of a Predator electro-optical sensor. Below each task are metrics to assess the accuracy, timeliness, and completeness to which the task was accomplished. For this type of analysis to work these metrics must be clearly and unequivocally measurable. Thus the metric should not be "how well did the operator do finding convoys?" but rather "was the convoy found?" or "was the convoy correctly identified?"

Note that the metrics we use in the ISDS laboratory are focused on the operator. This environment is not focused on technology evaluation; metrics concerning, for instance, algorithm cueing performance are not the primary concern. Assessing performance on the basis of metrics requires the collection of specific data. But to provide an in-depth analysis capability with the full fidelity to study and characterize operator performance, as well as discover unexpected artifacts in the process, the simple rule is "record everything." For our experiments, data collection services are deployed to gather

1. Operator sensor performance (e.g., electro-optical sensor field of view versus time);
2. Stimulus to operator (e.g., mission instructions to UAV commander);
3. All sensor data generated, plus the position and orientation of the sensor platform;
4. All vehicles' positions and orientations (ground-truth data); and
5. Operator and observer comments during the mission.

Data Collection Architecture

The primary mechanism for data collection is a service-oriented architecture [22] consisting of Web services that monitor the network (or local hard drive) for the arrival of new information. We refer to these Web services as "sniffers." Sniffers are configured for each network interface, network protocol, and port (or socket) in the computer operating system that carries data relevant to the simulation. Similarly, file-system sniffers collect data about files that are created or modified during the course of the experiment. In both cases, sniffers route the collected or discovered data to one or more data parsers. Parsers are another set of Web services tailored to a particular data format specification. They are designed to extract the important elements from the data stream, or provide metadata about the stream and store it in a searchable repository such as a database for

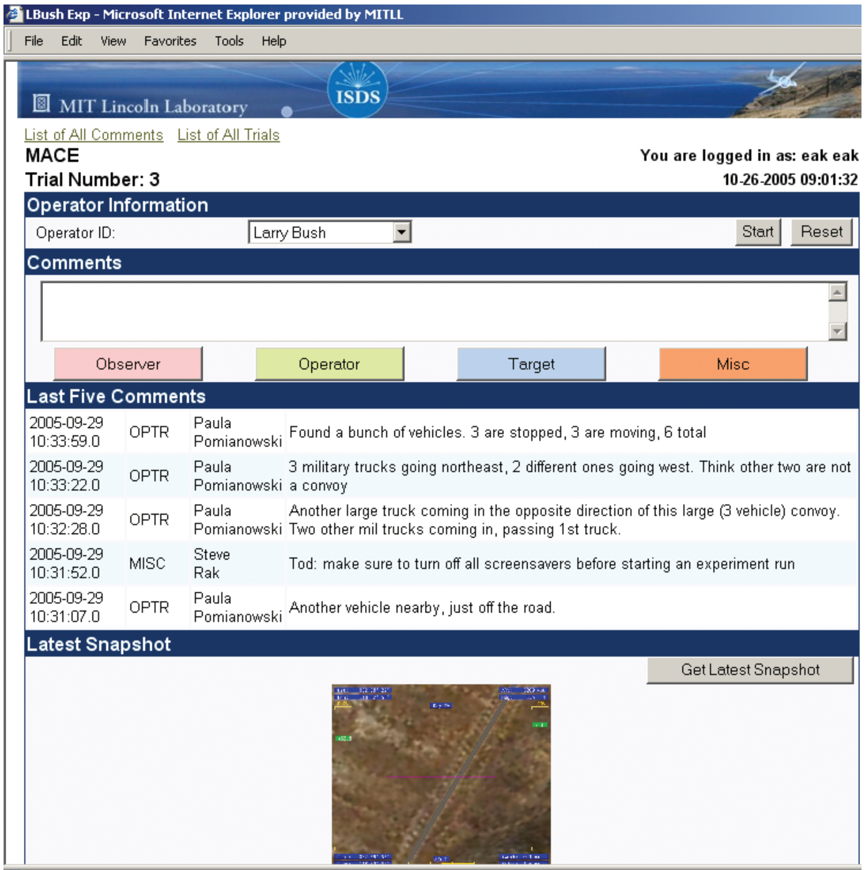


FIGURE 7. The Virtual Hammer observer/controller interface enables the controller to document inputs that are time tagged within the context of the experiment. These inputs are dynamically stored in a metadata database for after-action review and processing.

real-time or after-action analysis. In our modular design, multiple parsers may process the same input data for different purposes. One benefit of this service-oriented approach to data collection is its nonintrusiveness. We mentioned earlier that one of our goals is to provide data collection sidecars for ISR-based experiments. We are also using the Virtual Hammer simulation environment as a mock real-world feed to initiate our first level of capability testing.

This data collection environment is controlled from a common Web-services-based framework, which starts and stops the services and dynamically updates the status of each sniffer and parser—such as Run/pause or Packets collected. Additionally, our current version makes it possible to deploy parsers on the fly by placing a Java Archive (jar) file into the deployment directory; the Web application then dynamically loads the new parser, enabling its use without requiring a restart of the Web server.

In parallel, we use a Web-based observation form to collect and archive experiment participants' mission reports and comments. The observer/controller interface shown in Figure 7 enables the experiment controller to document inputs that are time tagged within the context of the experiment. Archived commentary might range from an observer's remarks regarding the operator's proficiency with the task or comfort with the test environment to critiques about how well the experiment is going, or ideas to try later to make things better. The Web form tags the comments in several broad classes to simplify later searching and analysis.

Once the data collected have been updated to a database, they are available for analysis—either as an after-action analysis to score the operator's performance, or in real time, using the information to provide feedback to the operator. This latter case provides the most direct avenue for prototyping decision support tools within the context of the ISDS laboratory. It puts the tools in front of the operator whom they are designed to assist, in an environment where the operator's performance and effectiveness are already under analysis. This approach thus provides the opportunity for gathering measurable feedback about a specific tool's ability to augment the operator's mission.

The MTI Activity Cueing Experiment

The inaugural Virtual Hammer operator-in-the-loop experiment was designed to illustrate the benefit of a wide field-of-view sensor, such as the SMTI component of a Joint Surveillance Target Attack Radar System aircraft or a future space radar constellation. The experiment could thus provide area-of-interest cues to a narrow field-of-view sensor, such as the electro-optical sensor on a Predator UAV. The main benefit to human operators of Predator's sensor is that it provides real-time target identification in areas otherwise unavailable for constant monitoring. However, the scarcity of the platforms, combined with their limited mobility, indicates an obvious need to provide cues on where best to conduct their surveillance mission. Wide field-of-view sensors such as SMTI are obvious choices for providing these cues.

The MTI Activity Cueing Experiment (MACE) was conceived as a way to measure operator-machine performance in ISR missions. Expectations were that novice and expert operators would display different behavior, a distinction that would help guide the development of decision support aids. MACE focused on the operator of the Predator sensor, measuring and capturing the simulation and sensor operator decisions with the eventual goal of evaluating the potential impact of a variety of sensor cueing strategies and decision support tools.

MACE follows a spiral development methodology, with each spiral increasing capability or complexity. The desire is to measure operator response to incremental differences in capability. Initially the sensor operator performed the mission in the absence of external direction, such as sensor cues. Using a measure of performance such as time to find a target bounds the problem space with an unenhanced measure of performance. Later, using the inherent truth of a simulation environment, we tested a perfect cueing capability to further bound the problem space and measure the operator's best possible performance. Succeeding spirals represent incremental capability in target detection algorithms and cueing strategies. The ability to collect all the simulation truth and operator's responses provides the opportunity for real-time feedback and post-mission analysis, relating measures of performance to measures of effectiveness such as successful and timely completion of the mission.

Experiment Components and Operator Mission

Virtual Hammer is well suited to support MACE because it provides the simulation environment that puts the operator in the loop. Virtual Hammer provides the background environment of the San Clemente Island terrain (including benign vehicle clutter as well as threat vehicles), the LiMIT radar platform flight path and its SMTI and SAR sensor collection schedules, and the out-the-window UAV view from VRSG (simulating the Predator EO camera). The Predator platform route was simulated as well, relieving the need for a pilot to fly the platform and freeing the experiment team to focus on the decisions made by the sensor operator alone.

The ISDS focus of MACE on measuring the Predator sensor operator's performance requires a mission in which the operator needs to perform a task and make a decision to complete the assignment. In this first phase, the operators were assigned a convoy detection mission. Convoys of threat vehicles were inserted with random start times and initial coordinates during the mission, and the operator was to locate and identify them as the Predator orbited the island. The operator would pan and zoom the sensor to search the mission area and, upon finding a potential convoy, verbally call out the sighting along with a count of the vehicles present and record a computer screenshot of the targets.

The Predator-sensor operator console views shown in Figure 8 consists of VRSG in Predator mode (simulating the electro-optical camera) and an in-house situation awareness (SA) display based on OpenMap [23]. The SA screen displays the Predator location and the sensor field of view. It also displays the area-of-interest cues from the SMTI operator or algorithm. When an operator selects a cue, it automatically slews the EO sensor to center the camera on the specified coordinates.

The experiment observer/controller also served as a surrogate UAV commander. While this functional role is not militarily accurate, it served laboratory needs to provide experiment support and observation in the person of someone who could motivate or redirect the operator, as needed. Typically this individual provided the verbal opportunity to report information. In the future

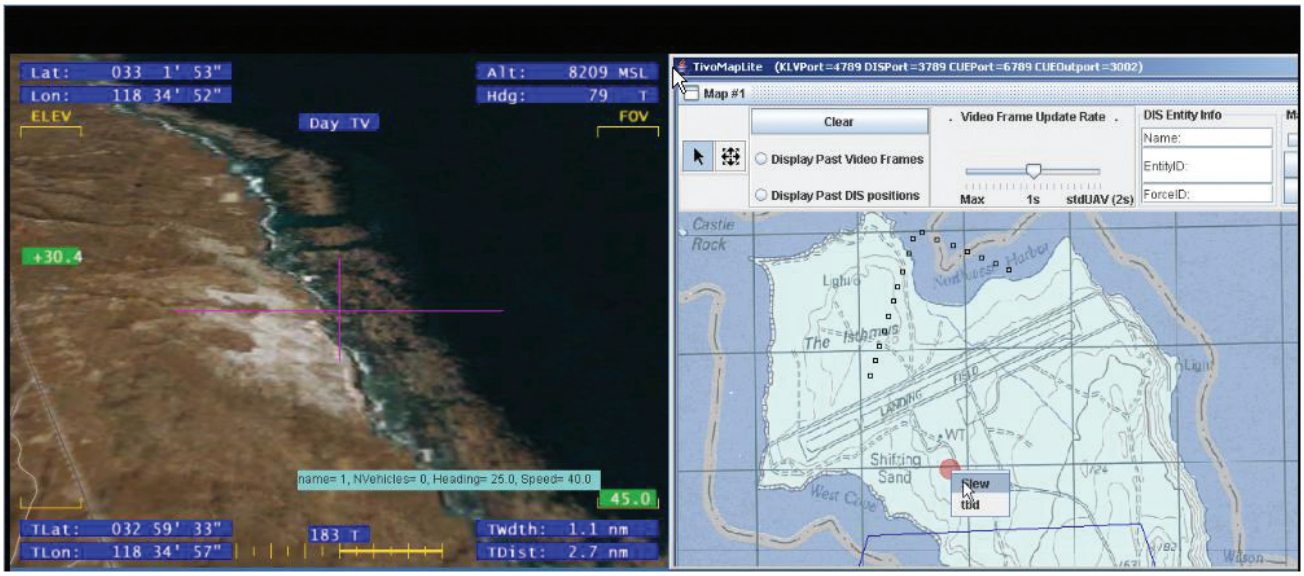


FIGURE 8. Screen views for the operator of an unmanned aerial vehicle (UAV). At left, the dynamic, interactive 3-D simulation of the Predator video display; at right, a simplified mock-up of the Predator situation awareness display, showing where the Predator has flown and where it's currently looking. This display also presents tips and cues from the convoy detection algorithms.

we will probably move toward a chat-based commander-to-operator interface, further insisting the chat is collected, time tagged, and archived with the other data.

Experiment Design

The MACE experiment is the product of numerous planning sessions modified by the early experience of several component-level and system-wide integration tests. The original vision had been to train and test large numbers of Lincoln Laboratory staff as operators to obtain statistically relevant data-quantifying enhanced operator performance. The goal was to use these results to pique the interest of military operators to participate in a more operationally relevant experiment. Because of the potential for statistical artifacts from unequally trained operators, it was decided that a small select group of operators would be trained on the sensor operator interface and mission concepts of operation. These operators would become familiar with the VRSG interface by performing practice missions to locate and identify landmarks in the simulation environment. Also, it was decided to randomly select the order that operators performed the cueing experiments (unaided, perfect cued, algorithm cued). These choices were taken to avoid learning-based artificial gains in operator performance.

Experiment Results

Figure 9 provide a glimpse into the types of results we have collected thus far in our initial MACE Spiral 1 (unaided cueing) experiments. The chart shows the results from two operators (one shown in red, the other in blue) as they attempted to find convoys. Operator search strategy was monitored by tracking sensor parameters such as image field of view (a measure of performance) versus time. The chart highlights differences between the operators in the number of convoys found

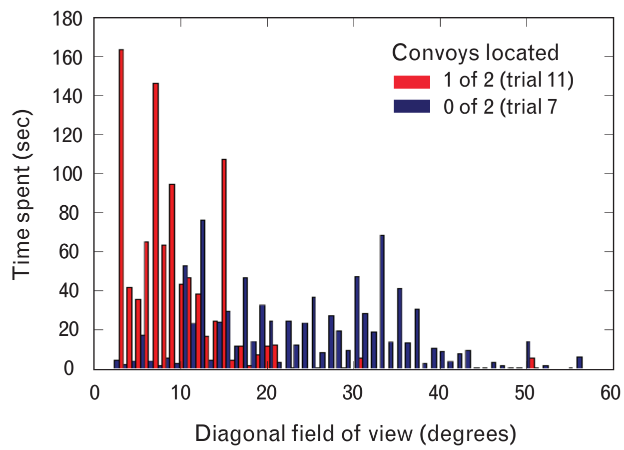


FIGURE 9. Initial results from two MACE Spiral 1 trials, contrasting a search pattern between two different operators. The operator shown in red spent most of his time at three discrete fields of view, and found one of the two convoys. The operator show in blue kept zooming in and out, and found neither convoy.

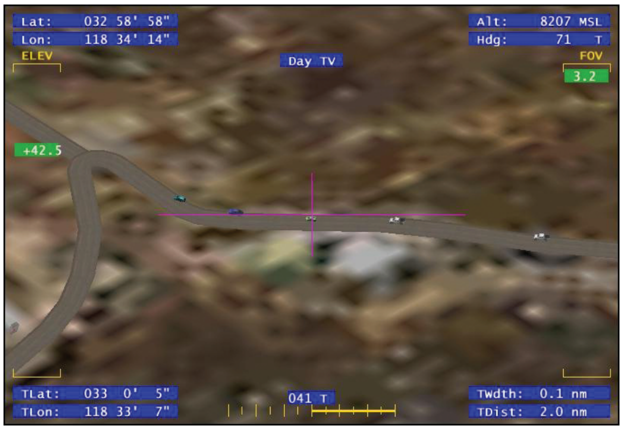


FIGURE 10. Snapshot taken by a UAV operator, showing a convoy of five vehicles within his field of view. When the operator detects a convoy (versus background traffic) he counts how many vehicles are in the convoy and saves a screenshot via a joystick button. The observer/controller can immediately view this screenshot and take note of whether the operator has correctly identified a convoy.

(a measure of effectiveness). Figure 10 shows a convoy as seen by a UAV operator.

Using the number of convoys detected as a scoring metric, we have developed statistical analyses to identify effective search strategies. The more effective operator (red) appeared to do the search mission at three particularly effective electro-optical sensor fields of view or zoom levels, detecting one of the two convoys. The less effective operator zoomed in and out, never really stabilizing into a disciplined search pattern, and detected no convoys. While this is an observation based on untrained civilian researchers and in no way is applicable to military trained operators, it provides an indication of the type of results that may be discovered upon analysis of the measures of performance.

Decision Support Tools

The results from the initial analysis conducted on the MACE uncued experiment led to a proposal to use this measure of performance as a guideline for electro-optical sensor operators to search more effectively. The analytical tool was proposed to be recast as a decision support tool. An example of this tool is shown in Figure 11. This prototype was developed in Matlab, using the same software that accesses the database for analysis. The difference is that the Matlab instantiation runs in real time. These developments illustrate the path by which decision support tools may be developed from insight gained during analysis of collected data.

The conclusions drawn about most effective sensor field of view or terrain covered to thoroughly search the area can easily be prototyped in a tool, and the conclusions retested for validity.

Excursions beyond Silent Hammer

The Silent Hammer exercise provided a good basic scenario from which to draw the requirements for the virtual environment. It included air, land, and sea assets linked by multiple communication systems in a controlled environment. Good information about systems performance and other ground truth for both live players and the natural environment existed and was readily available. While this exercise provided the data required to evaluate the virtual world performance vis-à-vis the real world, other much more challenging environments are of interest to the defense community.

Urban Combat—Baghdad

Urban operations are among the most unpredictable and dangerous that the military performs. Being able to develop and evaluate new methods for carrying out urban operations in a simulated environment would therefore be a huge win for the military. While simulations at various levels exist ranging from real-world-like excursions at the National Training Center to simulated urban tactical-team trainers, nothing adequately simulates the density of people, information, and objects (such as cars and trains) in a large city such as Baghdad (Figure 12).

The methods for tasking surveillance and other assets and analyzing the results are difficult to develop and risky to try out. Urban areas such as those in Iraq are often low-density/high-demand areas of ISR sensor utilization (large numbers of users competing for small numbers of sensors); consequently, ISDS research into ways of having multiple sensors cooperate to provide the best possible coverage would appear to have high value. Similarly, developing scheduling algorithms that have the sensors cooperate to minimize blind spots of coverage in urban canyons would be highly relevant to today's operational environment.

Coping with the thousands or potentially tens of thousands of dynamic, interacting objects in such an environment is an ongoing area of research in the simulation community. Virtual Hammer needs to move toward such a large and rich environment. Tomorrow's

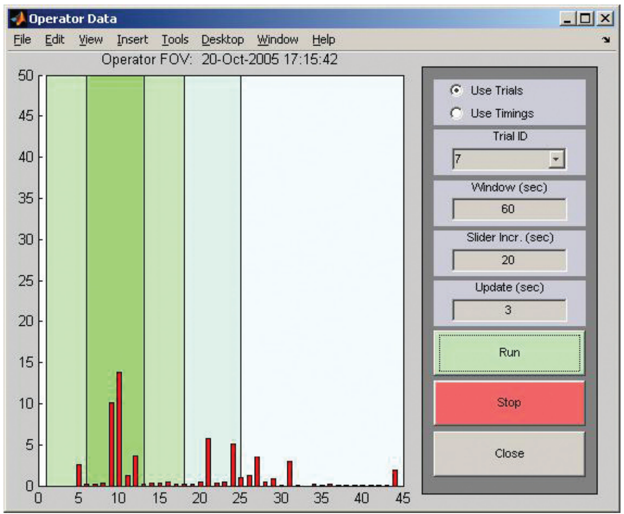


FIGURE 11. Prototype user interface for a real-time operator assessment tool. The concept is to present the observer/controller with an analysis of the operator's search pattern as the experiment is under way. This could further be adapted as a decision support tool for surveillance operators, providing insight into their search performance.

decision makers are likely to be overwhelmed by the vast sea of information that will be available to them from a new generation of wide-scale, persistent surveillance systems unless substantial progress is made in the area of decision aids.

Consequence Modeling

In the real world, operators care about their decisions and actions because they have a meaningful and long-lasting effect on people or places that the operators care about. If the operators view the environment simply as a consequence-free game, their actions and decisions may differ enough from the real world to invalidate the simulation. Early Virtual Hammer experiments have been pure surveillance: the operators perceived no ill outcome when they failed at their task. Negative operator performance could also be tied to negative simulation events, such as failed detections leading to the detonation of an improvised explosive device and destruction of (simulated) friendly vehicles or personnel.

It is essential that the operators become invested in the game in some meaningful way—emotionally, intellectually, or otherwise. This investment can vary from operator to operator and from scenario to scenario. Aligning appropriate incentives so that operators act in realistic ways will be a key factor in building a complete and credible simulation environment as the scenarios grow in size and complexity.

Acknowledgments

We thank the rest of the Virtual Hammer development team, including Jeff Allen, Bill Ledder, Tim Schreiner, Nagabushan Sivananjaiah, Rajesh Viswanathan, and Andy Wang. Special thanks to the Silent Hammer team, in particular Gary Condon and Paula Pomianowski. Thanks also to Captain Dave Duryea, our sponsor; to John Bilbruck of NAVAIR TSD for development of the San Clemente Island database; and to our industry partners, in particular Dave Hallforth of Camber, Brian Bilick of MäK, and the staff of MetaVR.

FIGURE 12. Simulated 3-D imagery of downtown Baghdad. This view is from the eastern shore of the Tigris River (seen in the upper left) looking northwest.

REFERENCES

1. OpenFlight Scene Description Database Specification Version 16.1, Document Rev. A, Oct. 2005, Multigen-Paradigm, www.multigen-paradigm.com/support/dc_files/of_spec_16_1a.pdf.
2 ISO/IEC 13818, "Information Technology—Generic Coding of Moving Pictures and Associated Audio Information," International Organization for Standardization, Geneva, Switzerland.
3. NITF (MIL-STD-2500) reference library, http://164.214.2.51/ntb/baseline/1999.html.
4. STANAG 4607: NATO Ground Moving Target Indicator Format (GMTIF), Edition 1, www.nato.int/docu/stanag/4607/4607_home.htm.
5. J. Calvin, A. Dickens, B. Gaines, P. Metzger, D. Miller, and D. Owen, "The SIMNET Virtual World Architecture," *Proc. IEEE Virtual Reality Annual Int. Symp. 1993, Seattle, 18–22 Sept. 1993,* pp. 450–455.
6. M.J. Zyda, D.R. Pratt, J.G. Monahan, and K.P. Wilson, "NPSNET: Constructing a 3D Virtual World," *Proc. 1992 Symp. on Interactive 3D Graphics, Cambridge, Mass., 29 Mar.–1 Apr. 1992,* pp. 147–156, 228.
7. IEEE Std 1278.1-1995, "IEEE Standard for Distributed Interactive Simulation—Application Protocols," IEEE, New York, N.Y., 26 Mar 1996, http://ieeexplore.ieee.org/xpl/standardstoc.jsp?isnumber=10849.
8. IEEE Std 1516-2000, "IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Framework and Rules," IEEE, New York, N.Y., Sept. 2000.
9. B. Sterling, "War is Virtual Hell," *Wired* **1** (1), 1993, www.wired.com/wired/archive/1.01/virthell_pr.html.
10. R. Richbourg, "Simulation for Reconstruction: Transformational Operations Produce Victory at Mazar-E Sharif," panel session, Interservice/Industry Simulation & Education Conf. (I/ITSEC), 2004.
11 SLAMEM/GVS, Toyon Research Corp., www.toyon.com/slamem_gvs.asp.
12. MäK Technologies VR-Forces, www.mak.com/products/vrforces.php.
13. Di-Guy, Boston Dynamics, www.bostondynamics.com/content/sec.php?section=diguy.
14. G. Benitz, "Adaptive SAR Results with the LiMIT Testbed," Adaptive Sensor Array Processing Workshop (2005), www.ll.mit.edu/asap/asap_05/pdf/Presentations/23_Benitz_P.pdf.
15. VRSG, MetaVR, Brookline, Mass., www.metavr.com/products/vrsg/vrsgoverview.html.
16. G.W. Stimson, *Introduction to Airborne Radar,* 2nd ed. (SciTech Publishing, Inc., Mendham, N.J., 1998), pp. 142–145.
17. Ibid., p. 147.
18. Ibid., pp. 147–148.
19. E.A. Alluisi, "The Development of Technology for Collective Training: SIMNET, A Case History," *Hum. Factors* **3** (33), 1991, pp. 343–362.
20. Camber Sensor Systems Division, www.cambertx.com.
21. P. Cho, H. Anderson, R. Hatch, and P. Ramaswami, "Real-Time 3D Ladar Imaging," *Linc. Lab. J.* **16** (1), 2006, pp. 147–164, www.ll.mit.edu/news/journal/pdf/vol16_no1/16_1_8Cho.pdf.
22. H. He, "What Is Service Oriented Architecture," O'Reilly XML.com, webservices.xml.com/pub/a/ws/2003/09/30/soa.html.
23. OpenMap Open Systems Mapping Technology, BBN Technologies, www.openmap.org.

APPENDIX: AUTOMATING THE EXPLOITATION OF RADAR DATA

The operator-in-the-loop experiments conducted in the Integrated Sensing and Decision Support laboratory measure the impact that decision support aids have on operator performance. Key ingredients of these decision support aids are automated data exploitation algorithms for radar intelligence, surveillance, and reconnaissance (ISR). Such algorithms aid and augment the operator by automatically performing many rote and otherwise time-consuming tasks and by focusing their attention on the most useful ISR data products. While algorithmic automation alone is not the answer to the information overload problem besetting human operators, when used appropriately as part of a decision support aid it can go a long way towards enhancing operator performance.

In the MTI Activity Cueing Experiment, for instance, a decision support aid helped unmanned aerial vehicle (UAV) video camera operators locate and identify vehicle convoys. An automatic convoy detection algorithm was developed with surface moving target indicator (SMTI) data collected during the Silent Hammer field experiment. This algorithm involves finding clusters of SMTI detections that are likely to be convoys, then associating these clusters over time by using a motion model.

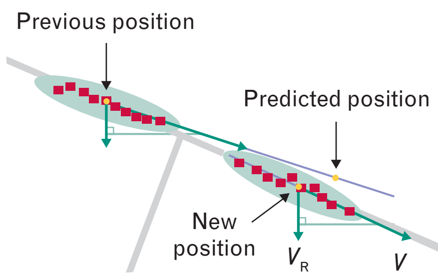


FIGURE A. Convoy motion model. The cluster line-of-sight velocity V_R is projected onto the estimated cluster heading to estimate the cluster speed V. The predicted position of the cluster from the previous scan and the position of the cluster from the current scan are used to associate the candidate convoys.

An SMTI detection includes the target position and the line-of-sight velocity from the target to the sensor. We can construct an accurate motion model from a cluster of detections by fitting a line to it, then projecting the line-of-sight velocity onto this line. We then use this motion model to associate clusters through time (Figure A) in order to accumulate evidence and dismiss false convoys. Ultimately, we identify the persistent true convoys by applying a threshold to this evidence. The flowchart of this convoy detection algorithm is shown in Figure B.

The Lincoln Multi-Mission ISR Testbed (LiMIT) radar data from Silent Hammer has large cross-range error and significant clutter, making convoy detection difficult. To combat these effects, we developed two filtering techniques that remove false detections by using SMTI feature patterns. We also reduce the cross-range error by taking into account topographic characteristics. These filtering techniques improved the subsequent clustering and association algorithm steps, thus improving the convoy detection accuracy.

In the future, we plan to use the Virtual Hammer environment to conduct experiments featuring decision support aids that enhance operators' ability to perform situational inference. Such aids could be powered by a class of algorithms we are currently pursuing, known as Bayesian multi-site situational inference (BMSSI) algorithms. These algorithms combine information from multiple sites in order to infer higher-level (e.g., regional) conclusions based on correlations in the data. The first step in BMSSI is to extract activity indications from sensor data. These indications can be as concrete or as abstract as desired; many examples stem from SMTI radar, since this sensor modality provides such a direct indication of activity, namely, movement [23].

Sensor modalities other than SMTI radar can provide indications of activity. For instance, significant changes in a particular area, as evidenced by a series of SAR images, can also be used as an indication of activity. As with SMTI, the primary advantage of SAR over infrared sensors is its search rate and its day/night, all-weather capability. Possible applications include monitoring ports, shipping lanes, air fields, military garrisons, rail stations, factories, weapon storage facilities, and other sensitive sites where the arrivals and departures of vehicles, vessels, and other large objects (e.g. cargo containers) could conceivably be detected by rapid revisit SAR (where "rapid" implies revisit times on the order of ten minutes to one hour).

To achieve that end, we have incorporated SAR change detection algorithms into our suite of tools we use to extract indications of activity from different sensor data sets. In preparation for change detection, the SAR images are rectified onto a topographic map, correcting geometric distortion. This step helps the co-registration of SAR images taken from different angles and thereby enables wide-angle SAR change detection, and making possible bimodal (e.g., SAR versus electro-optical) comparisons.

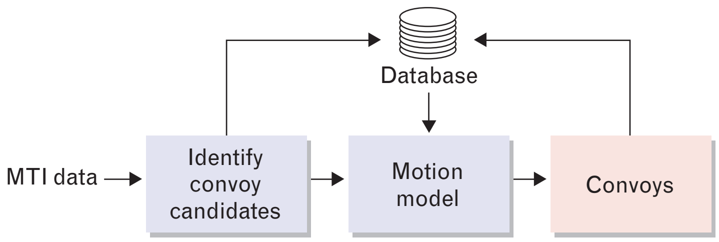


FIGURE B. The convoy detection algorithm for surface moving target indicator (SMTI) data. Clusters of SMTI detections likely to be convoys are used to construct motion models that underlie the convoy candidate association process illustrated in Figure A. Candidates that pass the association criterion are declared to be valid convoys.

Each of these activity indicators can be viewed as a separate node on a directed acyclic graph. BMSSI then proceeds to deduce the correct graph representation (Bayesian network) from the data by applying a greedy hill-climbing algorithm multiple times and using randomly selected starting graphs (Figure C). Once an appropriate graph structure is found, the parameters are estimated. The parameters are the conditional probabilities between the nodes. For example, one network parameter would be the probability of observing a particular value for Site Activity 1 conditioned on the state of the entire region.

Once the model has been trained, it can be used to combine information to infer higher-level (regional) conclusions. Specifically, we query the network to determine the probability that the region is in a particular state. The model can also infer the likelihood of seeing the activity at a particular site, in light of the activity occurring at all of the other sites. This is a form of anomaly detection that inherently accounts for extrinsic factors such as time of day or weather.

The above techniques are appropriate for allocating and cueing additional sensor resources to where they can be used most effectively. As part of decision support aids, these techniques present pertinent information to the operator in a concise form. They draw the operator's attention to activity of interest and provide vital sensor tasking information. As such, they are a key enabling technology toward unlocking the full potential of the rapidly growing U.S. military surveillance capability.

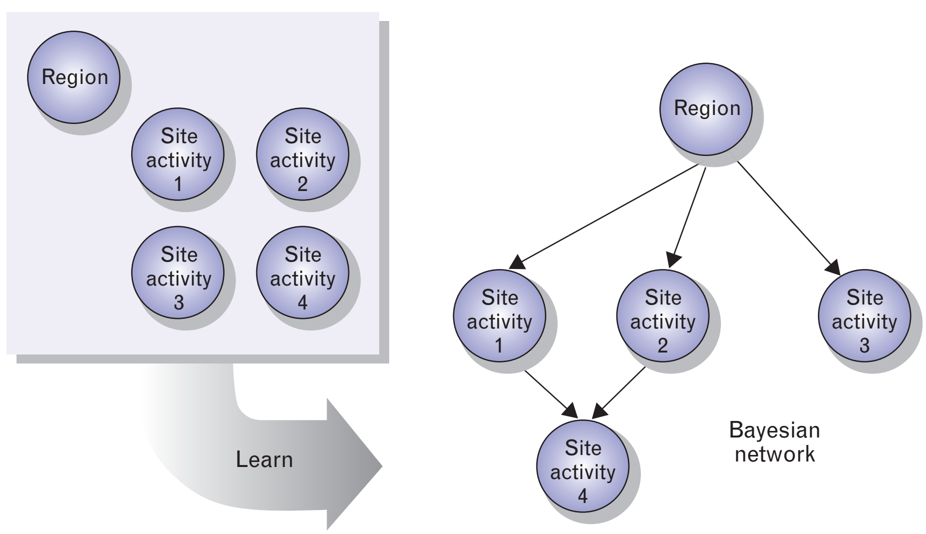
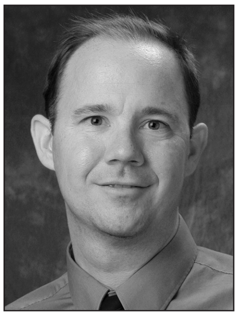


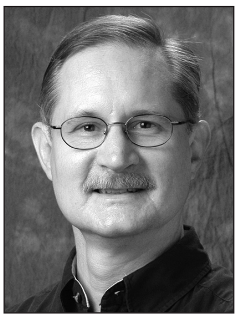
FIGURE C. Bayesian-network structure search. The structure of the network describes what influence, if any, the various observables have on one another. By starting with randomly selected initial graphs, then trying different random "mutations" on the graphs' topologies, an algorithm converges upon the simplest network structure that best fits a given set of training data.

PAUL METZGER
is a technical staff member in the Integrated Sensing and Decision Support (ISDS) group and was responsible for Virtual Hammer's original concept, system architecture, and detailed technical design. He previously led a team developing a variety of risk reduction modeling and simulation tools for the space radar program office, and was the senior system engineer responsible for the design, installation, and integration of the Laboratory's Boeing 707 into an airborne battle management center. He has a B.S. degree in electrical engineering from Northeastern University, and is founder of Reality by Design, the company that produced the first officially certified High Level Architecture (HLA)-compliant federate.



LAWRENCE BUSH
is an associate staff member in the ISDS group. His current research is in operations research and artificial intelligence techniques for automated decision making. He has a B.S. degree in industrial engineering and operations research from the University at Buffalo and an M.S. degree in computer science from Rensselaer Polytechnic Institute. He previously worked on expert systems at Cornell University, New York State Agricultural Research Station.

PETER MASTROMARINO
is a technical staff member in the ISDS group. His research interests include data mining and pattern exploitation, anomaly detection, image coregistration for purposes of change detection, and SAR-SMTI signal processing techniques. He has an A.B. degree in physics from Princeton University and a Ph.D. in physics from the California Institute of Technology.



STEVEN RAK
is a technical staff member in the ISDS group. His current work focuses on ISR system instrumentation and measurement of human-in-the-loop decision processes during exploitation of surveillance data. He received his B.S. and M.S. degrees in chemical engineering from Case Western Reserve University.

TOD SHANNON
is a subcontractor in the ISDS group. His background is in software development on distributed interactive simulation systems and 3-D graphics. Since receiving a B.S. degree in information and decision systems from Carnegie Mellon University, he has held positions at several companies, including Bolt, Beranek, and Newman; SAIC; and SensAble Technologies, and was a cofounder of Reality By Design.