

Reducing Re-Training Costs through On-Demand 'ad hoc' Assessment

Christian Sebastian Loh, Ph.D.
Virtual Environment Lab
Dept. of Curriculum & Instruction
Southern Illinois University Carbondale
csloh@siu.edu

I-Hung Li, M.S.
Virtual Environment Lab
Dept. of Curriculum & Instruction
Southern Illinois University Carbondale
henry.dea@gmail.com

Abstract. Although Multi-User Virtual Environments (MUVE) have proven valuable to training organizations and the military in mitigating training costs, first-generation MUVE tends to focus heavily on training and very little on evaluation and performance assessment (EPA). For effective EPA, existing data analysis methodologies need to be re-examined, and new methodologies developed, along with technology advancement to better assessment learning with virtual environments. It is best to correct man-made mistakes right away, than to delay until the training session is over – many hours later. Repeated and reinforced mistakes, if not caught in time, eventually became entrenched and required costly re-training for corrections. Traditional evaluation methodologies that took place only *after* (*post hoc*) the completion of training exercises are inadequate for just-in-time (*ad hoc*) assessment. This paper presents an on-demand, *ad hoc*, assessment framework and software tool for MUVE training that will allow trainers to monitor trainees' actions in real-time for correction and performance improvement.

1.0 INTRODUCTION

The recent advent rise of digital games and virtual worlds has offered much potential as a platform for virtual training. One common feature among these technologies is that they all take place within some sort of Multi-User Virtual Environments (MUVE). Besides the ability to automate mundane training tasks like other computer-based instructions, MUVE are also capable of co-locating massive number of trainees simultaneously, and thus, helping to mitigate training costs. The ability to train at a lower cost but with much greater features and capabilities is what makes MUVE appealing to training organizations.

According to Chen & Michael (2005), assessment is what sets serious games apart from other entertainment games. Hence, a suitable MUVE designed for training must provide trainers with a means to assess the trainee's learning. This would include monitoring the progress of learning, tracking the number of met objectives, identifying mistakes committed by the trainees, and allowing for appropriate remediation to be prescribed in a timely manner. Digital game-based learning and

training using MUVE could all benefit from a robust set of methodologies for evaluations and performance assessment (EPA).

1.1 Measuring Learning

Since there is no safe way to put a probe into the mind of a learner (regardless of the learning environment) to directly measure the amount of learning that occurs, trainers must rely on external indicators for EPA, such as test scores, classroom participation, time-on-task, and others. Within a physical face-to-face environment (e.g., traditional classroom), trainers can observe trainees' physical behaviors directly as evidence of learning and participation (Harrington, Meisels, McMahon, Dichtelmil-ler & Jablon, 1997). Unfortunately, the observation and measurement of human actions, behaviors and expressions directly within MUVE has, thus far, proven to be difficult.

1.2 Improving Performance

Performance improvement in the workplace (virtual, or not) is always about reducing waste and increasing output. Wastes, in this case, comprise of habitual man-made mistakes or errors, which can be very costly to unlearn via re-training. To increase

output, both trainers and trainees must learn to recognize mistakes (pointing them out as they are being made) and replacing it with the correct behaviors, so as to reduce the incidents of (costly) human errors.

Repetition is a core feature found in the majority of (educative) digital games in which users accumulate essential skills (or meet educative goals) through “trials and errors and repetition of steps” in order to progress in the game (Saridaki, Mourlas, Gouscos, and Meimaris, 2007). Depending on the scale of the MUVes, some training could last as long as 20-40 hours, spread over several weeks. After that long of a period, any unchecked error is likely to have become entrenched through reinforcement. Due to the massive numbers of trainees involved in training using MUVE, this problem will quickly reach critical mass!

2.0 EVALUATION & PERFORMANCE ASSESSMENT STRATEGIES

Since it is not practical to co-locate trainers in a 1:1 ratio with trainees within a virtual environment, trainers will require help in monitoring trainees’ actions to ensure that errors or inappropriate responses do not go undetected and become habitual. Instead of focusing on ‘training’ alone, a good MUVE will not only need to track trainees’ progress throughout the training sessions, but also provide ways for trainers to analyze the trainees’ data for EPA purposes.

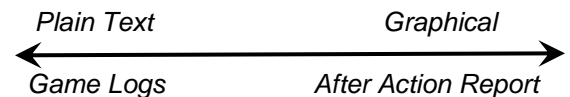
Unless “serious game” developers become more aware about the needs for EPA data collection, they will see no reason for the addition of ‘assessment components’ to MUVE. Some may perceive the ‘additional programming work’ needed to facilitate EPA as unnecessary, while others see them as costly overheads. Fortunately, the situation is improving, as a number of serious games now shipped with assessment components (Chen & Michael, 2005). Most, if not all, currently available ‘assessment’ employs a ‘*post hoc*’ strategy – i.e., they occur ‘after the fact’ (when MUVE training has been concluded). Currently, the most commonly

used *post hoc* methods are: (1) game logs and (2) pre-test/post-test design.

2.1.1 Game Logs

The Serious Games Showcase and Challenge (<http://www.sgschallenge.com>) is an annual serious games competition that attracts entries from commercial game developers, federal teams, and students alike. In the hopes of increased awareness for the importance of EPA, the organizing committee decided to include ‘assessment’ as a judging criterion in 2008. Since then, there has been a steady improvement in the quality of the game submitted, as well as an increase in sophistication of how ‘game stats’ are collected and presented for EPA.

The range of ‘assessment components’ seen in these submitted game entries may be represented by a ‘continuum’ (shown below) with simplistic game logs (i.e., ‘raw’ data in plain text or XML format) on one end, and graphical (fully formatted) After Action Reports (AAR) on the other.



Among the information collected by these assessment components, ‘timestamps’ of various game events (e.g., when objectives are issued or met) are the most common. Additional genre-specific meta-data may also be collected: e.g., orders of procedure being executed (as seen in many ‘medic’ simulations), total number of ‘objectives’ met (e.g., number of ‘kills’ in first person shooters, total number of cases solved), and the percentage of ‘coverage’ (e.g., total areas visited, total number of people spoken to, or the total number of missions completed).

Such information has typically been used as self-evaluation by trainees, as review logs of training sessions, or even as debriefing talking points among trainers and trainees. One common feature among these reviews is that they are all *post hoc* reviews carried

out after missions are completed. While *post hoc* debriefing sessions have their places in training, there are other situations when error corrections should be made immediately rather than be delayed. Some man-made errors, if left unchecked, can be reinforced to the point of entrenchment (via repetition), eventually requiring expensive retraining to unlearn. Many such examples can be found in literature about learning and reinforcement: e.g., language acquisition (Bybee, 2006), organizational behaviors (Luthans, 2002; Kreitner & Luthans, 1991), auditory learning (Loh, 2007), machine operation (Feibleman, 1996), and leaning-by-doing.

Unfortunately, game log by itself is rather low in utility to training organizations because majority of the trainers would not have the skills to analyze massive data sets. The transformation of these 'raw data' into consumable knowledge is often accompanied by high cost to the training organizations, because of the extensive manipulation and expert interpretation involved in the process.

2.1.2 Pre-Test / Post-Test Design

Other serious game researchers have, instead, chosen to conduct pretest-posttest experiments to assess the learning that occurs within MUVE. In these cases, a pretest is typically administered prior to the use of MUVE, and followed by a posttest after the training. The *delta* (Δ), or the difference in achievement scores ($t_2 - t_1$), is then accepted as indicative of the amount of learning that took place during the training with MUVE (e.g., Kebritchi, 2008).

However, this method of inquiry cannot fully explain which chain of events, or sequence of actions performed in the MUVE, actually contributes to learning. So, even if the experiment works, no one knows for sure how or why it worked. Thus, the MUVE remained an impenetrable "black box," making it impossible to detect any external factors (threats) that might have intruded

into the system and affected the data collected.

Without extensive monitoring and analysis, there is also no way to tell if trainees are trying to "game the system" (Baker, Corbett, Roll, Koedinger, 2008) – i.e., exploiting properties of the system to succeed in the environment rather than learning the materials as intended by the system designer. Unless trainees are quarantined (prevented from speaking with one another and accessing external learning materials), there is no way to ensure that the Δ reveals the actual amount of learning from MUVE.

Without knowing the factors that affect the success rate of the training, it would be difficult to convince training organizations to adopt a particular MUVE because success is at best, chanced. Furthermore, from the perspectives of teachers and trainers, the over-reliance on posttest data proves too unsettling because by the time the effectiveness of a learning module is determined (through *post hoc* evaluation), it may be too late and too costly to re-train. While this problem is not immediately apparent in 'clinical' research studies (with participants having 1-2 hours game play), the effect is often amplified in commercial off-the-shelf (COTS) games that require much longer (20-40) hours to complete.

3.0 INFORMATION TRAILS[®]

In order to overcome the 'black box' effect of the MUVE, it is necessary to track and measure user-data multiple times *during* the training itself using an *ad hoc*, or on-demand approach; instead of using an 'after the fact,' or *post hoc* approach. If it is at all possible, data should be collected *internally* using the game/software engine, instead of *externally* to avoid self-reporting and human input errors. Not only are trainers better informed by following the trainees' progress right from the beginning of the MUVE sessions, they are also able to catch mistakes made by the trainees earlier in the game/training and prescribe remediation accordingly.

The *Information Trails* is an EPA framework tailored specifically for training with MUVE (such as those commonly found in serious games and virtual worlds). Conceptually, *Information Trails* comprised of a series of event markers deposited at intervals within any information ecology (such as MUVE). Much like the trails left by Hansel and Gretel in the forest, once deposited, it would be easy for any investigator to 'follow the trails' to its destination. The framework was developed based on a series of research work involving online user tracking (Loh, 2006; Loh, 2007; Loh, Anantachai, Byun, & Lenox, 2007; Loh & Byun, 2009).

3.1 Data Visualization

The association of assessment with learning objectives is nothing new. The unique feature of *Information Trails* that sets it apart from the other EPA methodologies is that it presents the data collected just-as-it-happens via *ad hoc* reporting in a graphical format. Because the entire data collection and analysis procedure is now handled by the game engine and the *Information Trails* framework (automatically and discreetly), there is no need for trainers to try and make sense of the data collected – be it game logs or pre-test/post-test data.

Moreover, trainers could 'visualize' how and which learning objectives are being met within the context of the virtual training environment with the help of a data visualization software tool, in this case, a *Performance Tracer*. Should the training can be repeated over time, a pattern of behaviors for each trainee will begin to emerge from the *Performance Tracer* and can be subjected to further analysis.

3.2 Decisions > Actions > Behaviors

No matter the environment (virtual or physical), a person's actions and behaviors are ultimately determined by his/her decision making process. Using a game scenario, if a particular path leads to certain death (due to confrontation with a high-level

boss), players must decide if they will find alternative routes or be killed by the boss. Should they avoid the confrontation, they might gain the option to strengthen their characters and return at a later time for the challenge.

Repeated *actions* (to turn away from a path) will eventually yield an observable pattern of *behaviors* (to challenge the 'boss' only after sufficient training). This pattern of behaviors may then be inferred to reveal the person's decision making and reasoning processes.

Since a decision is the product of a person's knowledge schema, the effectiveness of a user's actions – speed, accuracy and strategy – within the information ecology may be expressed as a function of the users' understanding of the learning problems (what they know) and problem solving skills (what they are able to do).

4.0 FROM THEORY TO PRACTICE

In order to turn the *Information Trails* assessment framework from concept to a working prototype, a suitable development platform has to be identified. The COTS game known as *Neverwinter Nights 2* (NWN2, Bioware 2006) was selected because: (a) it came with a game development kit that allowed for easy modification, and (b) it allows for online play that requires user authentication. A third-party "Event Listener", called *NWN eXtender 4* (NWNX4), was necessary to act as a connector between the NWN2 game engine and the external database server.

Figure 1 shows the relationships among game engine, event listener, external database server, actionable learning and game objectives, and the *ad hoc* reporting system/tool. Apart from the game engine and event listener, all other components are in-house products. This includes the *Information Trails* EPA framework for MUVE-based training and the *Performance Tracer for NWN2*, a Rich Internet Application for data visualization (created using *Adobe Flex*). It should be noted that

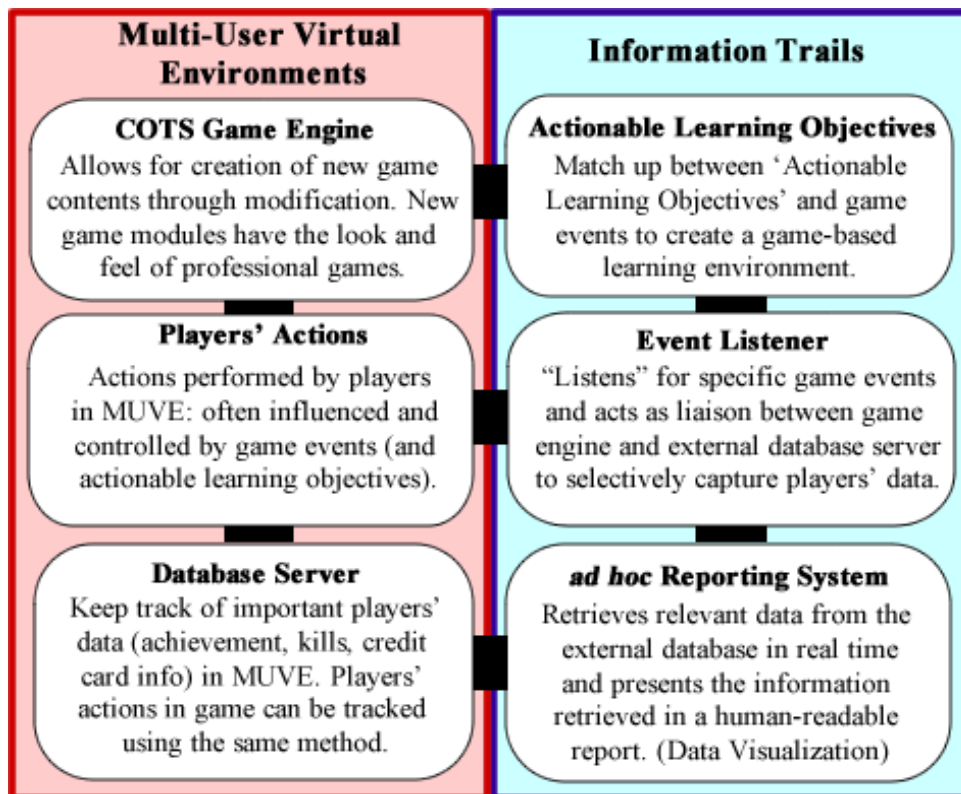


Figure 1. Game Environment and Information Trails Architecture

the *Information Trails* and *Performance Tracer* did not, by design, capture all available information indiscriminately. It is the authors' belief that choosing what data to capture is as important as what data not to capture.

4.1 Performance Tracer

In its current version, the *Information Trails* checks for learning-objective related events within the MUVE and captures key game events, including: game entry/exit, path traversed (movement), experience points gained, items gained/lost, module & area entered/exited, enemies killed, and conversation records. The data captured are then transformed into chunks of useful information and displayed graphically via the *Performance Tracer*. In EPA for MUVE, data visualization is an extremely important step in helping trainers and educators see the training "in-progress" via a human-friendly report, especially since not everyone is trained in handling vast amount

of data, or in interpreting what they mean.

Figure 2 (a box-and-line plot) shows the path traversed in one single training session by a player. Without additional (visual) information, the plot could not explain why the player traversed the training environment as such. Figure 3 shows the same path traversed superimposed over the area map. The map provided the much needed visual cues to a trainer to help him/her understand the reason behind the player's action/decision. Position markers (taken at 6-second interval) and event markers made up the series of connected boxes. The line connecting the boxes revealed an approximate path traversed by the trainee from Start (first box) to End (last box). The visual cues provided by the full-color bird's-eye view of the area (Figure 3) were important to training managers because they could finally match the trainees' actions to the geographical layout of the MUVE.

Other functions of the *Performance Tracer* included the animation of trainee's movements within the MUVE and "mouse-overs" that revealed various actions performed at the position markers. The interactive, real-time features showed above are simply not possible to display on paper-based/printed reports.

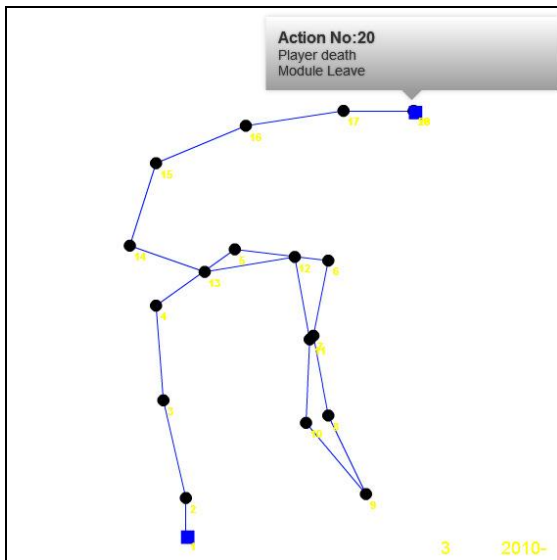


Figure 2. Record of path traversed by player within the MUVE. A 'mouse-over' of the event markers will reveal more information about the player's action.



Figure 3. Path traversed by player superimposed over an area map.

4.1.1 Standalone Viewer

It is possible that some of the above-mentioned reporting functions and animations may already be found in the latest high-end commercial off-the-shelf training games/simulations. However, it should be noted that *Performance Tracer* was never designed (in 2007) to replicate the AAR functions in COTS games. Our intention is to establish a new method of assessment using virtual environment, by standardizing the framework for users' behavioral data (and meta-data) collection in MUVE training, and further subjecting those data to data mining and visualization technique for better performance assessment and learning analysis.

This explains why *Performance Tracer* remains a standalone application independent of a game engine. The separation of the assessment reporting component from the game/MUVE allows trainers to retrieve trainees' data, anytime anywhere, without the need for an additional game license or installation. The reporting application can be adapted to allow any administrator in the reporting channel to check on the progress of the training, corporately or individually (per trainee).

4.2 Limitations

In order to reap the benefits of the *Performance Tracer*, it is necessary to incorporate the *Information Trails* framework at the level of game engine. In our example, since *NWN2* was not developed in conjunction with *Information Trails*, some of the game functions were too simplistic and limited for detailed behavior analysis.

For example, there were only two functions, *item_gained* and *item_lost*, to represent all possibilities involving the adding or removal of items from a player's inventory in *NWN2*. Even though players could gain items via any of the following means:

- obtained from a treasure chest
- bought from a merchant
- stolen from a non-player character (NPC)

- looted from a fallen enemy
- made by combining items (crafting) in the player's inventory
- created by a special spell
- given by an NPC, or another player in a persistent world

there was no way to truly differentiate the events from one another since they are all executed using the *item_gained* function.

Readers must understand that the economy for game development is very different from game EPA. From the point of view of the programmers, all seven possible actions were mere semantic differences that could be easily solved with the writing of one function: *item_gained*. Writing seven functions to represent each semantic possibility is additional (and needless) work for game programmers, regardless of the values they hold for training managers or *ad hoc* reporting.

There is an obvious need for collaboration between game developers and the developers of *Information Trails*, to integrate the framework at the game engine level, so that any game/MUVE written using the engine will have the benefits from both worlds. Retrofitting *Information Trails* into a completed MUVE is less than ideal, to say the least.

4.3 Future Development

Performance Tracer is a work-in-progress as we continue to improve the user interface and add new functionalities based on feedback received. For future development, we are looking to: (a) collaborate with game publishers to integrate the *Information Trails/Performance Tracer* framework at the game engine level, (b) consult with training organizations to enhance and customize the reporting and data analysis capability of the *Performance Tracer*, and (c) create a 'mobile' version of the *Performance Tracer* to facilitate performance review in the field.

5.0 CONCLUSIONS

The use of MUVE for training and instruction can revolutionize the way people

learn. However, the EPA tools used must be equally innovative. Ineffective EPA will yield questionable results, and may diminish the true value of the MUVE technology for learning and instruction. This is the reason why so many education technologies have been criticized to be "useless", "ineffective," and showing "no significant difference" in improving education (c.f. Cuban, 2001, and Clark, 2007).

Designing MUVE for learning purposes is very different from designing entertainment games because the former required the instructional designer to take into consideration the element of assessment, and the latter has no need to do so. Linda G. Roberts, ex-Director of Education Technology to the U.S. Department of Education, once said, "I believed that researchers could improve the design and collection of data. Just as new technology created new opportunities for learning, it created ways to invent new tools for research and evaluation, particularly ways to track and monitor what, how, and when learning occurred" (2003, p. viii).

New assessment methodology must keep pace with the advances of technology for MUVE, in order to provide educators with the assessment data needed to garner support from stakeholders for these innovative instructional approaches. For the serious game publishers, integration of *Information Trails* and *Performance Tracer* at the game engine level will not only add assessment capability to the company's flagship product, but also provide *ad hoc* reporting capability in all MUVE developed using that engine.

For the trainers, instead of waiting for the entire exercise to be over before debriefing, they can now communicate with the trainees' about their actions and decision making process in real-time, using the information revealed by the *ad hoc* reporting tool. For those who require *post hoc* review, the animation function (like an instant replay) of the *Performance Tracer* will be

useful in critiquing trainees' actions for performance improvement. The move towards a mobile viewer should ease deployment within training organizations. Man-made errors committed during training can be rectified before they run the risk of becoming entrenched, thus saving training organizations valuable time and money.

6.0 REFERENCES

- Baker, R. S. J. d., Corbett, A. T., Koedinger, K. R., & Roll, I. (2006, June 26-30). *Generalizing detection of gaming the system across a tutoring curriculum*. Paper presented at the 8th International Conference on Intelligent Tutoring Systems, Jhongli, Taiwan.
- Bybee, J. L. (2006). From usage to grammar: The mind's response to repetition. *Language*, 82(4), 711-733.
- Chen, S., & Michael, D. (2005). Proof of learning: Assessment in serious games. *Gamasutra*. Retrieved from http://www.gamasutra.com/features/20051019/chen_01.shtml
- Clark, R. E. (2007, May-June). Point of view: Learning from serious games? *Educational Technology*, 47, 56-59.
- Cuban, L. (2001). *Oversold and underused: Computers in the classroom*. Harvard University Press.
- Feibleman, J. K. (1996). Technology as Skills. *Technology and Culture*, 7(3), 318-328.
- Harrington, H. L., Meisels, S. J., McMahon, P., Dichtelmiller, M. L., & Jablon, J. R. (1997). *Observing, documenting and assessing learning*. Ann Arbor, MI: Rebus, Inc.
- Kebritchi, M. (2008). *Effects of a computer game on mathematics achievement and class motivation: An experimental study*. Ph.D. Doctoral Dissertation, University of Central Florida, Orlando, FL.
- Kreitner, R., & Luthans, F. (1991). A social learning approach to behavioral management: Radical behaviorists "mellowing out". In R. M. Steers & L. W. Porter (Eds.), *Motivation and Work Behavior* (pp. 47-65). New York, NY: McGraw-Hill College.
- Loh, C. S. (2007). Choice and Effects of Instrument Sound in Aural Training. *Music Education Research*, 9(1), 129-143.
- Loh, C. S. (2006). *Tracking an avatar: Designing data collection into online games*. Paper presented at the annual conference of the Association for Educational Communications and Technology (AECT 2006), Dallas, TX.
- Loh, C. S., Anantachai, A., Byun, J., & Lenox, J. (2007). *Assessing what players learned in serious games: In situ data collection, information trails, and quantitative analysis*. Paper presented at the Computer Games: AI, Animation, Mobile, Educational & Serious Games (CGAMES 2007), Louisville, KY.
- Loh, C. S. (2008). Designing online games assessment as "Information Trails". In V. Sugumaran (Ed.), *Intelligent Information Technologies: Concepts, Methodologies, Tools, and Applications* (Vol. I, pp. 553-574). Hershey, PA: Information Science Reference.
- Loh, C. S., & Byun, J. H. (2009). Modding Neverwinter Nights into serious games. In D. Gibson & Y. K. Baek (Eds.), *Digital Simulations for Improving Education: Learning Through Artificial Teaching Environments* (pp. 408-426). Hershey, PA: Information Science Reference.
- Luthans, F. (2002). The need for and meaning of positive organizational behavior. *Journal of Organizational Behavior*, 23(6), 695-706.
- Robert, L. G. (2003). Forewords. In G. D. Haertel & B. Means (Eds.), *Evaluating educational technology: Effective research designs for improving learning* (pp. 290). New York, NY: Teachers College Press.
- Saridaki, M., Mourlas, C., Gouscos, D., & Meimaris, M. (2007). *Digital games as a learning tool for children with cognitive disabilities: Literature review and some preliminary methodological and experimental results*. Paper presented at the European Conference on Games Based Learning, Scotland, UK.

7.0 ACKNOWLEDGMENT

This research is made possible in part through funding from the 2009 Defense University Research Instrumentation Program (DURIP) from the U.S. Army Research Office, and research assistantship from the Southern Illinois University Carbondale.