

Mixed-reality Environment Advanced Threat Simulator (MEATS) Project

Teja Akella

New Albany High School, New Albany, Ohio

take11a6315@gmail.com

Abstract— The objective of this paper is to outline the work done for the GRILL MEATS project over the Summer of 2022. The project goals for this time frame were to create a functional Live Virtual Construct (LVC) simulation that is optimized for Rocket-propelled grenade training, to create a virtual RPG that utilizes Vive Trackers and has virtual Reality (VR) capabilities, to have the capabilities of both sending and receiving Distributed Interactive Simulation (DIS) data to allow for a multi-player interactive simulation and to create a 1:1 replica of the training facility site. This paper primarily focuses on the three steps of developing this simulation: designing and programming a simulation that allows for Vive tracker integration and has VR capabilities, allowing for the simulation to actively send and receive data, and finally creating interactive 2D and 3D widgets that allows the user to turn on and off various settings, features, and scenarios.

I. INTRODUCTION

This project is being completed in partnership with AFRL, The Gaming Research Integration for Learning Laboratory (GRILL), and Wright State University's Calamitville, which is a training facility for special forces and first responders. The focus of this project is to create an LVC Simulation which is creating integration between virtual and real assets to enhance training, testing, and research. No matter what the industry or job some type of training is required. Air Force and Army soldiers have to undergo rigorous weapons training for weapons such as RPGs, 50-Cals, and other specialized weapons. Currently, the Air Force and Army physically train soldiers by making them fire actual RPGs. This system, however, has been determined to be inefficient because each shot fired from an RPG can cost upwards of \$3000 and it is unsafe for both the trainer and trainee. This current process is also very damaging to the environment as it can cause ozone depletion and global warming. The GRILL was tasked with creating a mixed-reality simulation that makes RPG training safer, easier, and more efficient, and also shows that LVC simulations are a better alternative to real-world training. The tools that we used to meet the goals of this research project were Unreal Engine 4 and its block-based coding and VR development environment, SteamVR to manage the Vive Trackers that were used to get position and orientation data, and the GRILLDIS Plugin which is a proprietary Air Force networking plugin that allows for the sending and receiving of DIS data in Unreal Engine 4. DIS is an IEEE standard and network protocol. It works by sending marked data up to the network with a unique entity ID. Then other computers that

are running the simulation will retrieve this marked data by looking at the data's unique ID. DIS is a popular simulation networking protocol that is used in the game design industry mainly for its features of allowing for multiplayer interactive simulations where multiple users can be in the same simulation. The hardware used for this project was a 1:1 sized model RPG, Vive tracker 3.0's, a Raspberry PI, a U-Blox 7 GPS sensor, a 9-axis iNEMO inertial module unit sensor, and a Vive Pro VR headset and controller that allows for VR capabilities. Vive trackers are a special type of tracker that sends orientation and position data to Unreal Engine. On the back of the Vive tracker, there are 8 pogo pins and they were used to wire a button trigger so that when a user is training they can press the trigger on the model RPG, and then in the virtual reality setting an RPG ballistic will be fired at the intended target. In addition to the Vive trackers, a Raspberry PI was used with 2 sensors so that it can replace the Vive tracker if necessary. The PI is very beneficial because the data that it gathers from its sensors are more accurate and therefore makes the simulation smoother. The two sensors that are interfaced with the Raspberry Pi gather GPS, Acceleration, Orientation, Magnetic Flux Field, and Temperature data. This data is then sent up to the network with a unique ID as mentioned before and then, Unreal Engine is able to retrieve this data by matching a hard-coded ID value with the data on the network.

II. METHODOLOGY

Creating the simulation had two major tasks that needed to be completed:

1. Learning about how to create a VR simulation in Unreal engine and how to integrate Vive trackers into that simulation
2. Learning how DIS works, the theory behind how DIS is sent and accepted, and how to implement it into Unreal Engine 4 and Raspberry PI

The first step was to build an environment in Unreal engine that was a 1:1 replica of the Calamityville site, as seen in figure 1. Once this was done, RPG physics was implemented into the simulation by spawning in an object with the texture pack of an RPG projectile, and then when the mouse button was clicked, the projectile would move at an approximate velocity of 294 m/s in the intended direction. Once the RPG physics was working with textures and an accurate detonation scheme, the Vive trackers were attached to the physical model

RPG and they were paired with SteamVR, as seen in figure 2. After pairing them, the trackers were added as motion controllers that track the orientation of the RPG in Unreal engine. After this, the VR headset was incorporated into Unreal engine by creating a VR Human that would act as the base operative at which VR would function. After configuring VR a scope-down sight was added to the RPG to give a realistic feeling of scoping in with a real RPG, as seen in figure 3. After giving the RPG a realistic look and feel, a button was hardwired to the Vive tracker to act as the trigger and it was placed on the actual trigger of the model RPG, which once again makes the simulation more realistic. After the RPG was tested and made sure of being accurate a Raspberry PI was interfaced with a GPS sensor and an Inertial Module Unit, as seen in figure 7, to get the position, Acceleration, Orientation, Magnetic Flux Field, and Temperature data. After all of this data was obtained, the Raspberry PI was configured with DIS so it would send data up to the network with a unique ID and then the simulation would retrieve it. As seen in Figures 5 and 6 this was the required logic to make Unreal engine send and retrieve DIS data from the network. After DIS was implemented, in order to make a more user-friendly simulation, an options menu widget was added. This menu had many features including settings for VR and trackers, networking information, scenarios, smoothing options, as well as weather options.

III. DATA AND RESULTS

This project has acquired complete functionality and has met all of its goals. It has a 1:1 exact replica of Calamityville's military training facility as seen in figure 1 and the RPG used for the simulation has been completely customized with trackers and buttons to ensure that the user has an almost natural training experience, as seen in figure 2. To further ensure a natural training experience a scope sight with 2x zoom has been added to give a realistic aspect to the simulation. Figure 4 shows the menu which has many features, scenarios, and networking information that the user can utilize to customize their experience as well as ensure that their training is of a high standard. Figures 5 and 6 show the completed code for DIS sending and receiving. The accuracy with which Unreal Engine 4 and the Raspberry PI send and receive were evaluated using Wireshark, which is a protocol analyzer and this analyzer showed that the data sent and received was 100% accurate. Finally figure 7 shows the Raspberry PI and the sensor integration, which was an alternative to the Vive Trackers. However, the GPS sensor that was used malfunctioned many times and because of this, the results of the physical location that is reflected in the simulation were off by many meters.



Fig. 1 Design of 1-to-1 replica of Calamityville Attachment



Fig. 2 Image of RPG Model with attached Vive Tracker



Fig. 3 Scoped Sight Attachment



Fig. 4 Menu widget attachment

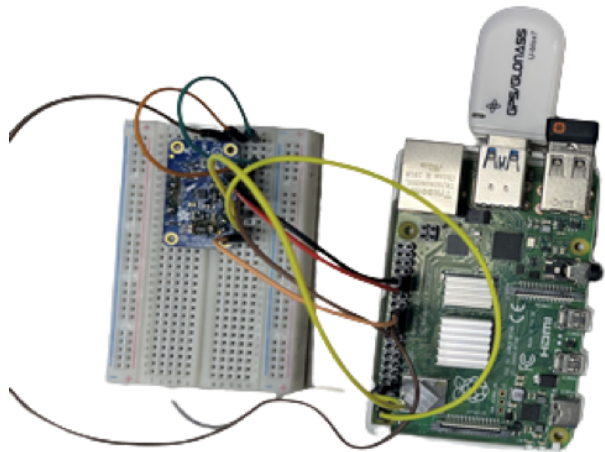


Fig. 7 Raspberry PI sensor integration

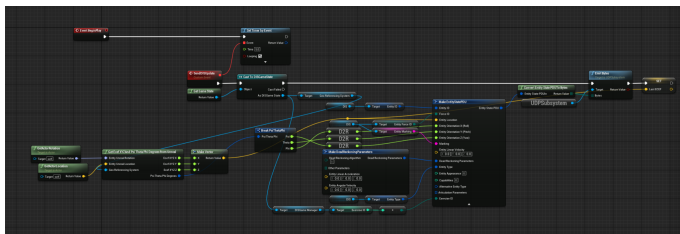


Fig. 5 DIS Block-Based Coding For Sending DIS Data

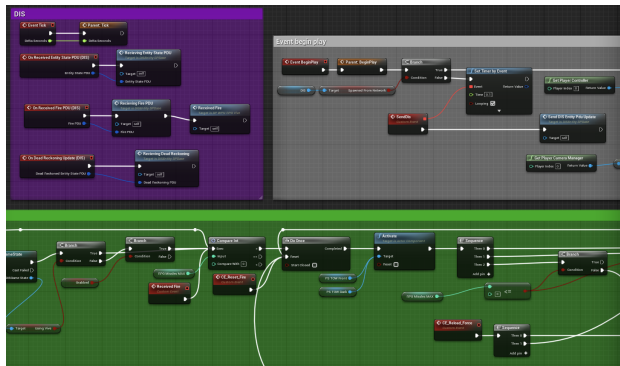


Fig. 6 Block-Based Coding For Receiving DIS Data

IV. CONCLUSIONS AND FUTURE STEPS

Overall, at the end of the project, the three overall goals were reached. The simulation allowed for RPG training and was able to interface with Raspberry Pi and Vive Trackers, and DIS integration. The next step for this project would be to make a feature that allows streaming to mobile devices, which we coined as a portal into the virtual world. Also, further optimizations and implementation of targets can be made as further improvements to the virtual Calamityville environment.

Description of Research:

This past summer, working at the Air Force Research Laboratory's (AFRL) most cutting-edge facility: The Gaming research integration for the Learning laboratory (GRILL), I gained immense hands-on experience in training simulation research. I worked with customers from Calamityville, a military training facility for special forces and first responders, to create a Live Virtual Construct (LVC) training simulation, which integrates real and virtual assets to enhance training, testing, and research. I used Unreal Engine 4, Python, cybersecurity networking, and Raspberry Pi sensor integration to devise a new method of virtual reality (VR), which is denoted as mixed reality, for our customers at Calamityville. This means that if an item in real life is repositioned, the object will also appear to move in a VR simulation setting.

Contributions to project:

My role in this project, as a research scholar, was to interface the Raspberry Pi module and the Vive trackers to Unreal Engine 4 so that they would allow for the sending and receiving of DIS data through the network. DIS is an IEEE standard and network protocol. It works by sending marked data up to the network with a unique entity ID. Then other computers that are running the simulation will retrieve this marked data by looking at the data's unique ID. DIS is a popular simulation networking protocol that is used in the game design industry mainly for its features of allowing for multiplayer interactive simulations where multiple users can be in the same simulation. This data was obtained from the Vive tracker and for the Raspberry Pi, the data was acquired by using sensors that obtained GPS, Acceleration, Orientation, Magnetic Flux Field, and Temperature data. All of this data was then coded to be packaged up and sent to Unreal engine where this data would be used to constantly update the telemetry data regarding the RPG.

What I learned about the research process:

I have learned a significant amount about the research process and how research is actually just trying new things and seeing if your perspective on something can make an impact on the current industry. I want to dive deeper and explore more possibilities while working with this proprietary technology because I realized that it allows for endless possibilities due to its high level of customization with training, and I was at the forefront of conceiving it. After this research experience, I have realized that my research interests do not align with a certain major or field because CS has such a wide variety of applications to many fields whether that be medicine, aerospace, or even linguistics. Therefore, I would like to conduct research in many different fields that use CS so that I can contribute to that research while enhancing my skills.