29.61

393882

p 6

1997

NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM

MARSHALL SPACE FLIGHT CENTER
THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

VIRTUAL ENVIRONMENT USER INTERFACES TO SUPPORT RLV AND
SPACE STATION SIMULATIONS IN THE ANVIL VIRTUAL REALITY LAB

Prepared By: Joseph D. Dumas II, Ph.D.

Academic Rank: Assistant Professor

Institution and Department: University of Tennessee at Chattanooga
                                      Department of Computer Science and
                                      Electrical Engineering

NASA/MSFC:

        Laboratory: Mission Operations Laboratory
        Division: Training and Crew Systems Division
        Branch: Systems Branch

MSFC Colleague: Joseph Hale

IX
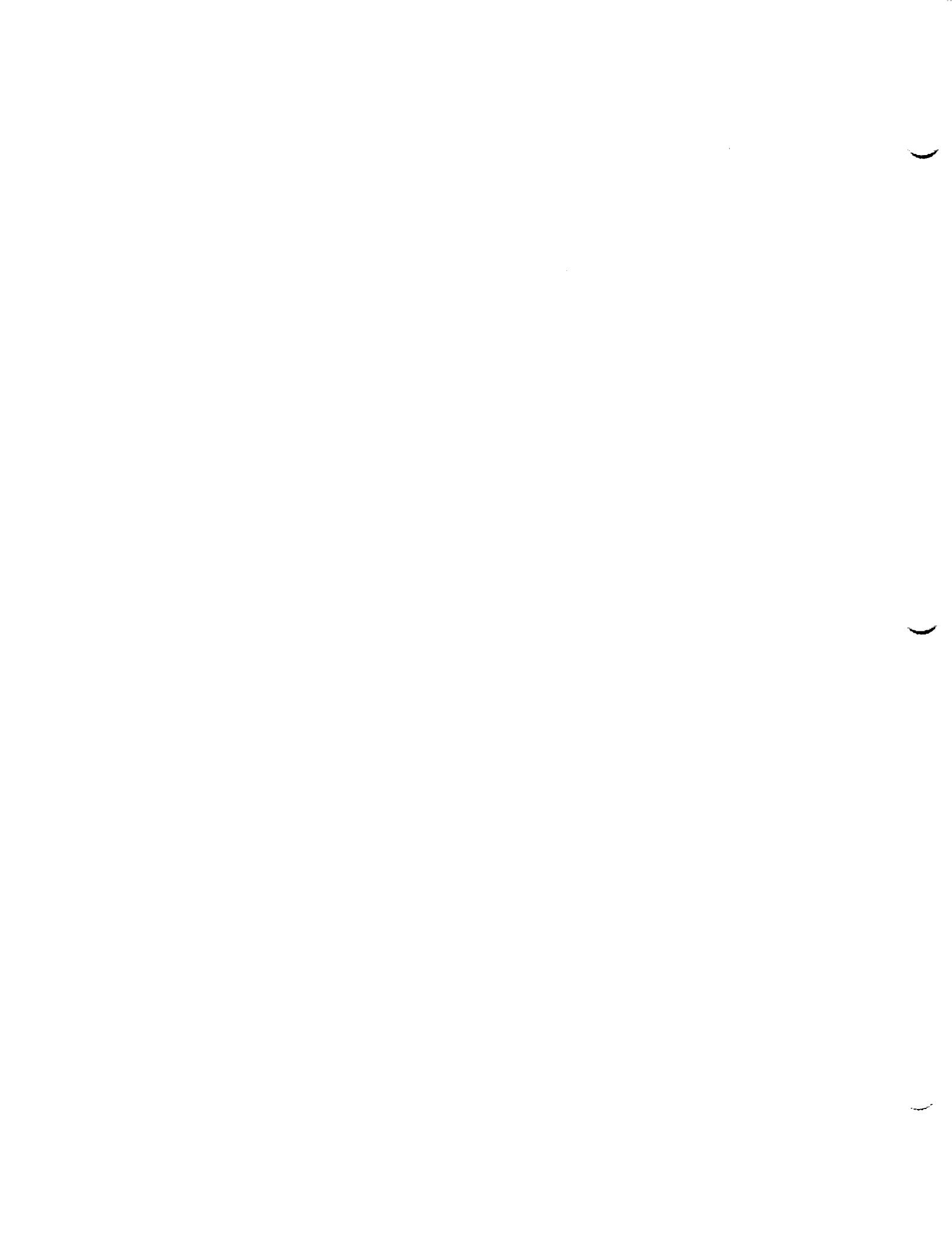
Introduction

The Army/NASA Virtual Innovations Laboratory (ANVIL) has been established at Marshall Space Flight Center to conduct investigations into the application of Virtual Reality (VR) technology. This new laboratory combines the resources of MSFC's former CAVE (Computer Applications and Virtual Environments) lab facility with those of the former Army Missile Command virtual reality program. As a result of this collaboration, the new ANVIL laboratory possesses a variety of hardware and software resources, including several Silicon Graphics computer systems and two software packages designed for developing and running VR applications. These packages, WorldToolKit (by Sense8 Corporation) and dVISE (by Division, Inc.), support a number of standard peripheral I/O devices which are used to allow a user to interact with the virtual environment. For example, both dVISE and WorldToolKit support magnetic trackers such as the Polhemus Fastrak which are used to track the movements of the user's head, hands, and/or other body parts, as well as other devices such as the mouse and Spacetec Spaceball which may be used to navigate in the virtual world. Both packages also provide support for display devices, such as the CrystalEyes stereographic viewing glasses (by StereoGraphics Corporation) as well as a number of head-mounted displays (HMDs), which allow the user to view his/her virtual surroundings.

Some of the peripheral devices in the ANVIL laboratory, for one reason or another, are not supported by one or both of the VR packages. This requires that custom hardware interfaces and/or software drivers be defined and put in place to allow data generated by the user's interaction with these devices to specify his/her interaction with the simulation and to allow the user to view the virtual world. As part of his Summer Faculty Fellowship, the author was tasked with developing and verifying as many of these peripheral interfaces as possible in order to support Reusable Launch Vehicle (RLV) design analyses and other ANVIL projects. The ultimate goal is to develop the capability and expertise required to produce VR simulations using any or all of the I/O devices with either of the software packages in support of any project assigned to ANVIL. In addition, some time and effort were expended to complete the integration of a networked simulation of the Space Shuttle Remote Manipulator System (RMS) being used for a Space Station assembly task, which was begun during the previous summer's Fellowship period.

Development of Peripheral I/O Interfaces and Drivers

The peripheral I/O devices that we desired to integrate with the virtual reality software packages discussed above fall, in general, into one of two categories: input devices, which are used to track user motions or accept navigational input for moving through the virtual environment, and visual output devices, which provide the user a view of his/her surroundings in the virtual environment. Some of these I/O devices, such as the VPL DataGlove, are not supported by Division or Sense8 because the manufacturer has gone out of business and/or the device is considered obsolete (even though functional devices exist at MSFC and many other sites). Other devices, such as the Fakespace FS^2, are so new, and installed in so few locations, that software support is not yet available. Finally, some peripherals, such as the Space Shuttle RMS hand controllers (present in the laboratory as part of the RMS simulation

discussed below) are nonstandard devices unique to NASA and of course are not supported by commercial software packages.

The hardware interface for the RMS hand controllers was designed and implemented during the summer of 1996 as part of the development of the Space Station module docking simulation described more fully below. This interface consists of a 486-based PC with a data acquisition card which reads analog data from the controllers, converts the data to digital form, and transmits the data over an Ethernet-based local network to a Silicon Graphics Indigo^2 which runs the WorldToolKit-based simulation. At that time, it was decided to configure the hand controllers so that they can be used either to control the RMS and its payload (their intended purpose) or, by sliding a switch, to "fly" the viewpoint (move the user within in the virtual world). This proved to be a judicious choice, as it was discovered that the RMS hand controllers work very well as a general-purpose navigational input device for virtual reality simulations. This prompted the author's development of driver code that can be compiled with any WorldToolKit simulation to allow the use of the hand controllers as a "sensor" device. A sensor can be "attached" to the user's viewpoint or to any object in the virtual universe in order to generate movement. The original version of this driver allows navigation in three dimensions (six degrees of freedom: x, y, and z translation and roll, pitch, and yaw orientation). At the request of one of the NASA personnel in the laboratory, an alternate version was developed that constrains the user to two-dimensional (three degree of freedom) movements in the x-y plane. The RMS hand controllers are now fully functional as a sensor device in WorldToolKit simulations.

The dVISE software that was acquired in early 1997 is a more full-featured development package than WorldToolKit. It is also more object-oriented and uses a Graphical User Interface (GUI) to make it easier for the end user to create VR applications. One of the first projects undertaken by the author this summer was to develop a driver that would allow the RMS hand controllers to be used with dVISE. This was complicated somewhat by the fact that dVISE applications are not constructed in C program code as in WorldToolKit, but rather can only access code which has been pre-compiled and linked into a specific shared library format which is not well documented in Division's on-line manuals. However, with some technical assistance from Division support personnel, the author was able to develop a hand controller device driver which creates a sensor that can be attached to any part of the user's virtual body. For example, the hand controllers can be coupled to the virtual head (in order to move the viewpoint) or to the virtual hand (in order to point, etc.). An enhancement to this basic capability uses the dVISE Virtual Keyboard (VKB) feature to generate "button presses" which activate events in the virtual world, in particular "picking" or grasping a virtual object. Finally, an enhanced hand controller driver was developed which mimics the action of two separate sensors (the gray button on top of the right hand controller acts as a toggle switch for selection of a sensor) so that two virtual body parts can be controlled. This allows the user to travel to a desired location in the virtual world by "flying" the head, then toggle to controlling the virtual hand so that a nearby object can be touched and, if desired, picked up and moved. This feature is very useful for interacting with many types of simulated environments.

Another input device which is very useful for picking up objects in the virtual world, and can be used even more intuitively than the hand controllers, is the DataGlove. This device senses the amount of light transmitted through optical fibers to characterize the bending of the wearer's fingers, allowing him/her to touch or grasp virtual objects or make gestures which have meaning in the virtual world. The ANVIL laboratory has two VPL devices, a "Glovebox" which controls a single DataGlove, and a Data Acquisition/ Transmission Unit (DATU) which allows the use of up to four DataGloves simultaneously. These devices were made by VPL Research, Inc., a firm which has been out of business for some time; thus, Division and most other software developers do not support the DataGlove. The author was able to find some old VPL example code for reference and, after some work, was able to establish data communications to and from the Glovebox via its RS-232 serial interface to allow calibration of the glove to a user's hand and retrieval of the finger bend data. Efforts to make use of the DATU were thwarted by the lack of documentation and insufficient time to "reverse-engineer" the special, nonstandard interface used by the device. The author's dVISE dataglove driver allows a user to grasp and release a virtual object by bending and straightening the forefinger. The bending of the forefinger generates button press events that cause an object to be picked up if it is being touched by the virtual hand. The object can then be moved by moving the hand, which is tracked by a Polhemus Fastrak sensor (for which Division does provide support), and released by straightening the forefinger. At present, the finger bend data are not used to articulate the virtual hand; however, dVISE provides the ability to do this and the finger bend data are available if this level of realism becomes necessary for a particular application. No WorldToolKit driver was developed for the DataGlove since that package does not render a representation of the user's body and thus there is no virtual hand with which to grasp objects.

Another device which is new to the ANVIL lab is the Fakespace FS^2 (also known as the LeBaron since the first unit was sold to Chrysler for use in designing that automobile). The FS^2 is a new product (MSFC's unit is the eighth installed worldwide) which is both an input and output device. It consists of a mechanical-linkage head motion tracker (which provides reduced tracking latency compared to magnetic trackers) coupled to a high-resolution stereographic HMD used to view the virtual environment. Since there are so few of these devices installed, there is as yet no software support from Division or Sense8. Fakespace did provide the source code for a generic test program that is used for all their products; this illustrated the use of some of their library functions which can be used to initialize the tracker and access the data for the user's head position and orientation. Using these library calls, the author was able to create drivers for dVISE and WorldToolKit which allow the head data to control the user's viewpoint in the virtual world. User presses of the two buttons on the HMD are also detected by the code and can be used to perform virtual actions as allowed by each software package.

With drivers for several input devices in place, some attention was given to configuring the dVISE software for use with various visual output devices. Division's package allows one to configure displays without writing any program code, but rather by making entries in a "registry" file which specifies the configuration details. In addition, some display devices require the issuance of operating system commands in order to alter display characteristics at the hardware level. At the start of the author's Fellowship period, the only

display devices usable with dVISE were the "mono" (flat system monitor screen) and CrystalEyes displays. With considerable experimentation, successful configuration of the Fakespace FS² and Virtual Research EyeGen3 displays was accomplished, giving the ANVIL lab the capability for true "immersive" virtual environment simulations.

Completion of Shuttle RMS Simulation

During the summer of 1996, the author helped develop a real-time, immersive simulation of the task of docking a pair of Space Station modules using the Shuttle RMS. This simulation was conceived not only with the idea of simulating space operations on earth but with the intent of demonstrating the feasibility of using high-resolution range sensors in combination with an immersive virtual environment as a superior alternative to video camera observation for on-orbit operations. For the purpose of demonstrating the concept, the range sensor (which would provide the relative offset and orientation of the modules being docked) was replaced with a high-fidelity, real-time simulation of the RMS and payload dynamics. This code runs on an Alliant FX-8 supercomputer in the Robotic Simulation Lab at MSFC. The integration of the virtual docking simulation thus required the networking of three computer systems (a personal computer used for hand controller data acquisition, the Alliant, and the Silicon Graphics host which renders the virtual environment). All hardware interfacing, data communications, and graphics programming necessary for integration of the simulation were accomplished during the author's 1996 Fellowship period; however, problems with integrating the RMS dynamics model prevented the successful demonstration of closed-loop operation. This summer, in cooperation with personnel from the Robotic Simulation Lab, the author was able to complete the integration of the simulation by successfully extracting the RMS joint angles, offset coordinates, and Euler angles from the dynamics model, applying the appropriate transformations, and coupling the data to the graphical representation of the RMS and payload in the WorldToolKit simulation. The simulation is now fully integrated and can be viewed either on the "mono" display or immersively by using the EyeGen3 HMD.

Summary and Conclusions

Several virtual reality I/O peripherals were successfully configured and integrated as part of the author's 1997 Summer Faculty Fellowship work. These devices, which were not supported by the developers of VR software packages, use new software drivers and configuration files developed by the author to allow them to be used with simulations developed using those software packages. The successful integration of these devices has added significant capability to the ANVIL lab at MSFC. In addition, the author was able to complete the integration of a networked virtual reality simulation of the Space Shuttle Remote Manipulator System docking Space Station modules which was begun as part of his 1996 Fellowship. The successful integration of this simulation demonstrates the feasibility of using VR technology for ground-based training as well as on-orbit operations.