Southern Illinois University
CARBONDALE

SIU-CAVE: Cave Automatic Virtual Environment

Brady Sprinkle, Josh Maier, and Utsav Dhungel

Mentor: Dr. Christos Mousas

Website: http://www2.cs.siu.edu/~jbu/

Introduction

A cave automatic virtual environment (CAVE) is a form of virtual reality that allows for an immersive, multi-user experience without limiting one's spatial awareness or mobility.

A typical CAVE system consists of:

- Five screens that form a cube
- Five projectors that project synchronized-stereoscopic images to the user(s)
- PCs networked together to run the application synchronously on all screens
- Input devices and motion sensors synced for localization and input signals

Purpose

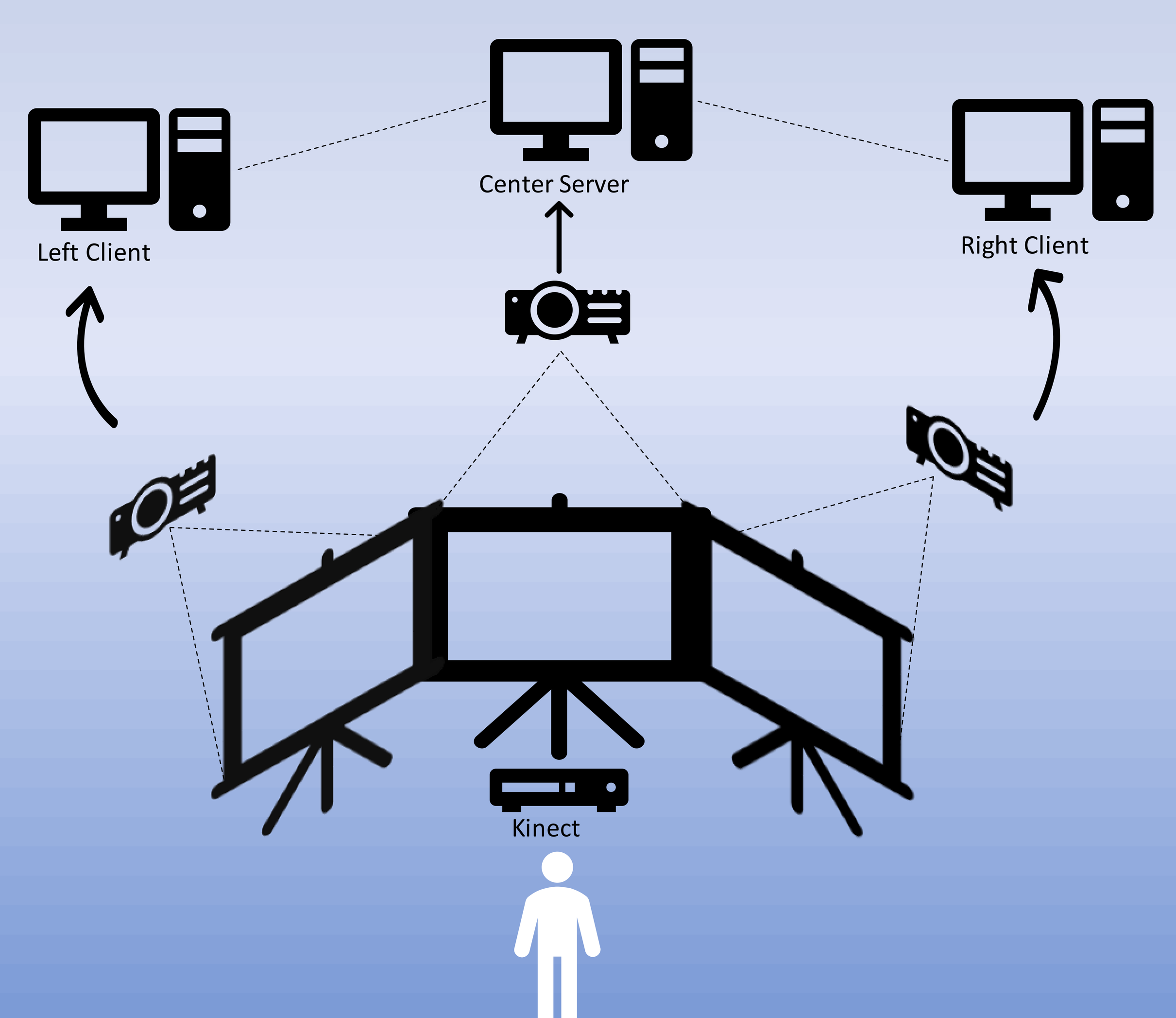
Create a prototype CAVE system to better analyze and demonstrate the features of CAVE's, as well as provide an initial platform for developing CAVE VR technologies at SIU.

High-end CAVE systems are incredibly expensive and complex, reaching upwards of $20,000. Our CAVE prototype aims to illustrate the benefits of these systems while being cost-effective.

Challenges

The complexity of the project stems from the compatibility and interaction of different types of hardware and software, including:

- Collecting and localizing the user's input data
- Synchronizing and calibrating viewports
- Determining and optimizing the user's field-of-motion and field-of-view



Methods

ARCHITECTURE DESIGN:

- 3 Screens constructed at 60° angles to form a C-shape
- 3 Projectors behind the screens at a distance of six feet
- 3 PC's connected to their own projector, and then networked together; two clients and one server
- Rear-projection screen material to eliminate shadows
- Screen and projector stands
- Kinect motion sensor placed underneath the middle screen and a Wiimote held by the user
  Total cost: ~$630

SOFTWARE DESIGN:

- Unity Game Engine for application development
- MiddleVR is a plugin used for camera calibration, networking, and input-device communication with Unity
- VRPN is a tracker for input data used with the Wiimote
- Kinect SDK: is a tracker for input data used with the Kinect

INPUT DESIGN:

- The Kinect tracks the user's yaw movement; when you turn left or right the cameras adjust accordingly
- Rotating the Wiimote modifies the user's Z and Y-axis independently; simulating moving forward, backwards, and fine-tuned turning
- A sense of localization is achieved while using both inputs

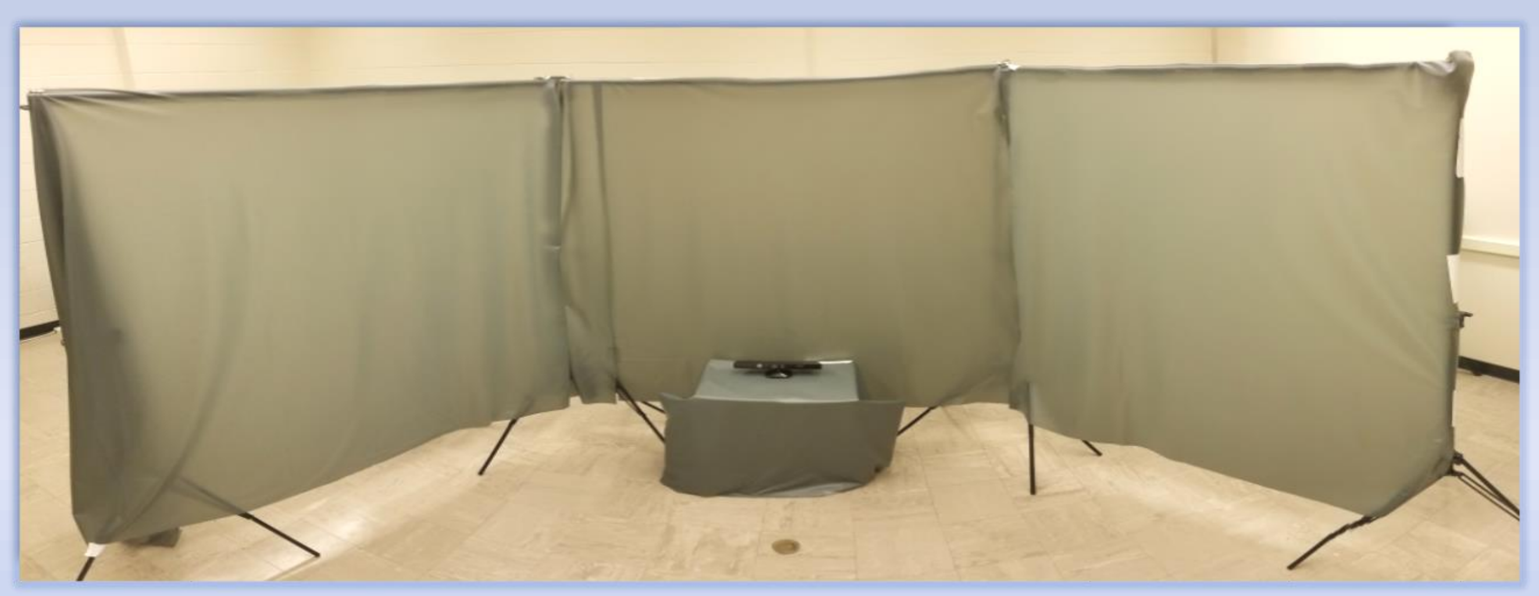
Development Environment

Unity Game Engine:

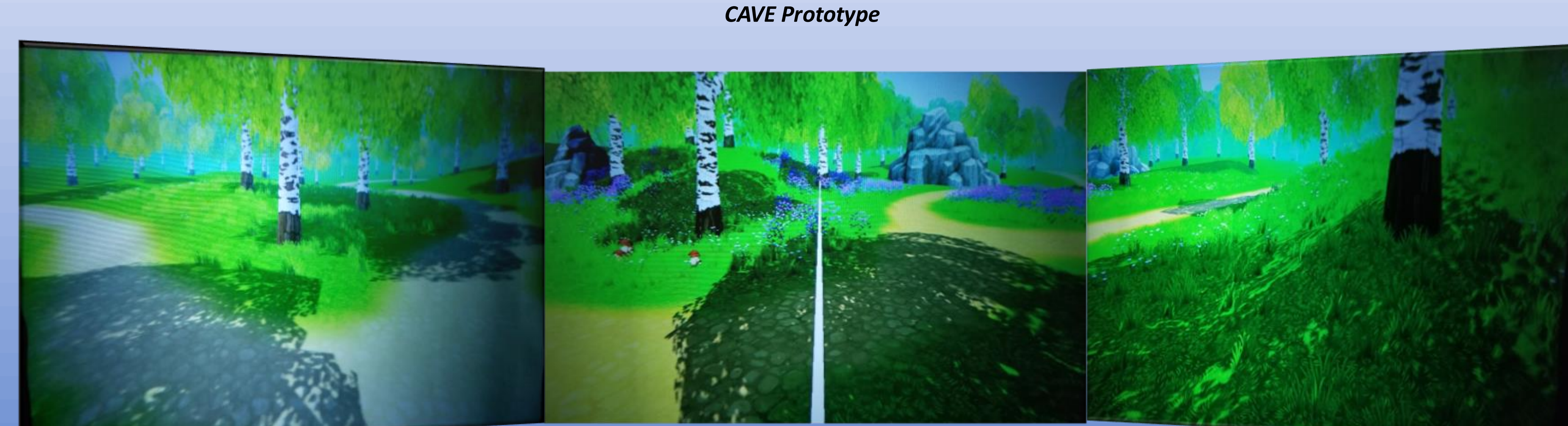
- Platform chosen for application development, compatible with MiddleVR plugin
- MiddleVR's plugin is used to further edit settings within the project

MiddleVR:

- Creates a configuration file with viewports, cameras and input devices settings.
- The configuration file is imported into the Unity application
- Cluster daemons run on each computer in order to synchronize instances
- An instance of each project is then ran on each machine via the network



CAVE Prototype



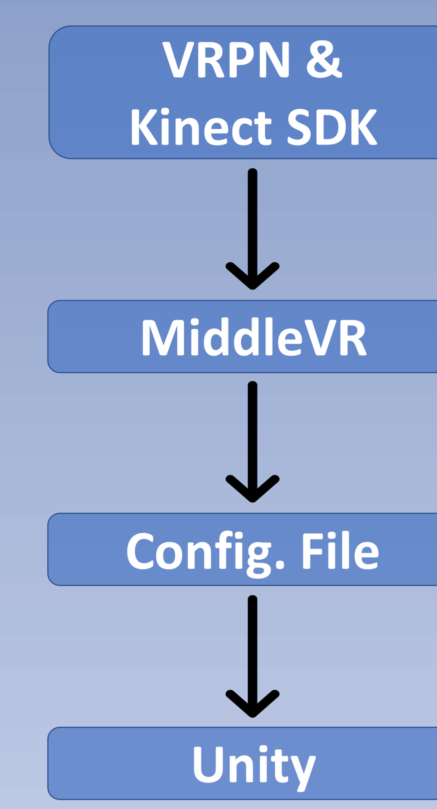
CAVE Demo

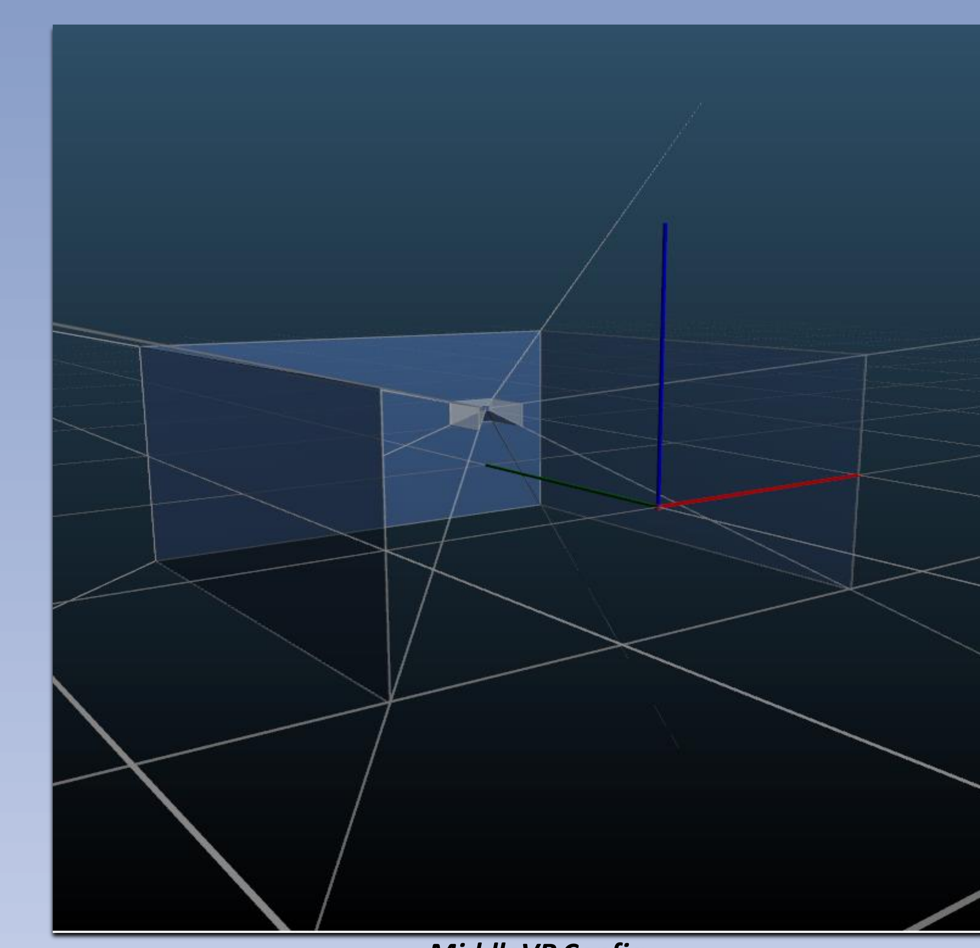
Discussion

The real value of a CAVE system is the variety of applications it can be used for. These are some of the current fields benefiting from CAVE technology:

- Automotive and aerospace engineering
- Medicine
- Architecture
- Education
- Media and entertainment
- Military
- Real estate

At SIU, an application for a CAVE system can be designed to utilize the multi-user functionality and the virtual environment. The CAVE can be used as a virtual lab that would enable students to practice high-risk scenarios that would otherwise not be feasible, such as medical operations.





MiddleVR Config

Conclusion

With limited funding, we successfully created an entry-level CAVE that demonstrates the key features of high-end systems. For developing the CAVE further, we recommend the following investments and adjustments:

- Additional screens for the floor and ceiling of the CAVE to improve immersion
- Develop SIU's own calibration software to improve CAVE-application flexibility
- A larger room for CAVE development minimizes calibration and field-of-motion issues
- Improve hardware
  - Projector's throw distance needs to be minimized for high-quality, large images
  - Screens need to be seamless, large, and of high quality to improve fidelity
  - High-performance GPU's for improved framerate
  - Input devices with three-dimensional tracking

Acknowledgements

We would like to thank the department for the budget to do our project as well as Michael Barkdoll and his GA's for their technical assistance. Also a big thank you to Professor Mousas and Professor Hoxha for their guidance and support!

Department of Computer Science, 1230 Lincoln Drive, Engineering A319, Mail Code 4511, Carbondale, IL 62901

Senior Project, Team JBU, 2017-2018