Virtual Reality Force Feedback and Safety Device

Team Members: Stephen Siemonsma, Sam Hisel, Kevin Mattes, George Drivas
Sponsor: Professor Tyler Bell

Abstract

Virtual reality headsets provide a user with an immersive visual experience which easily allows a user to lose track of where they are in a physical space. Headsets like the Oculus Rift provide a system to show a play space boundary in-game, but there is nothing limiting a user's arms from leaving the boundary. The goal is to create a force feedback device to limit the risk of the user accidentally moving their hands outside of their VR play area boundary. The device should be capable of a variable level of resistance of outward arm extension. These variable resistances can also be adapted inside the game for more realistic environment interactions.

Introduction

Current VR head-mounted displays all suffer from the same safety issue: users blinded to their surroundings quickly lose track of where they are. The inspiration for this device comes from the lack of physical feedback inside of a VR environment.

To solve this issue, a force feedback system needs to be created using two major design requirements:

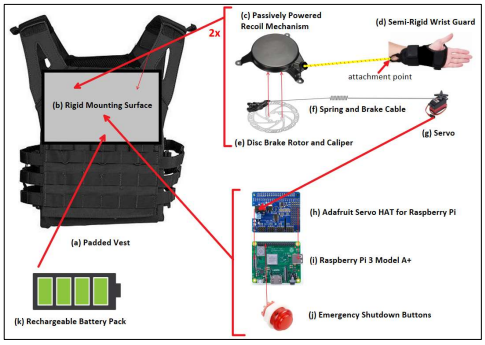
1. Restrict users from moving their arms outside the play area.
2. Provide a variable level of force feedback to allow for a realistic sense of collisions with in-game VR objects.

Oculus Rift headset with controllers

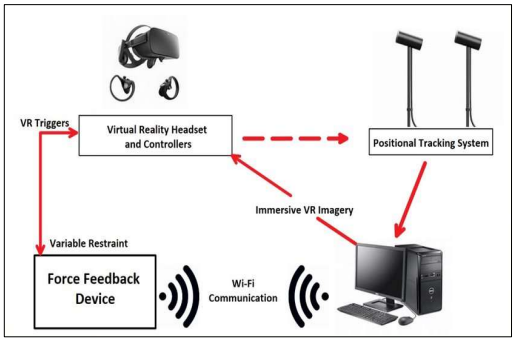
Aim

The aim of the project is to provide a safer and more immersive experience for current and future VR enthusiasts. There is a vast and untapped potential for force feedback devices in the VR industry, but it seems like upcoming products are overly focused on vibrational haptics and fine motor feedback. We are attempting to tap the potential of larger-scale force feedback interactions with the user's body. Our device can restrict the outward motion of the arms and simulate real impacts. This type of interaction allows our system to both protect the user and give them a taste of the future of virtual reality immersion.

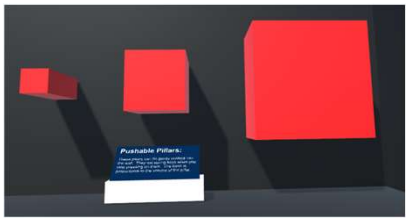
Methods

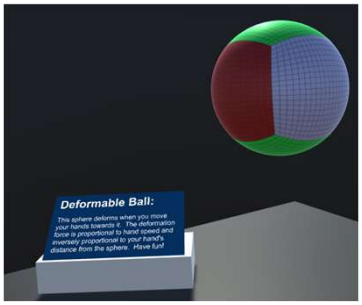


- Development of a mechanical device that provides a variable level of resistance to outward arm movement, based on input from Oculus Rift's positional tracking system
- Development of a networking protocol used between Unity and the force feedback device
- Utilization of positional tracking system and Unity collision physics to calculate an in-game force to be applied by the force feedback mechanism
- Creation of a virtual reality demonstration program to highlight the features of the system



Demo Environment Stations





Results

- Designed and created a wearable device to accomplish our goals
- Safety features include 3-dimensional play area boundary enforcement, vibrational alerts, safely enclosed physical components, an emergency stop button, wrist guards, fall detection, the option of a low-force game mode, and plenty of useful instructions for the user
- Acrylic front panel for easy viewing of mechanical components
- Variable level resistance device accomplished using a disc brake controlled via a servo and passive recoil mechanism
- Made use of the UDP protocol for finer control over network latency and performance
- Utilized Unity's physics engine for in-game demonstration of device

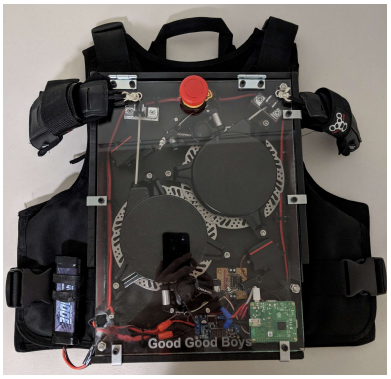
Conclusion

Our device is very helpful for VR users who want to play in small spaces or who want a more immersive experience while also maintaining their own safety and protecting any objects surrounding the desired play space. Using our system, users will be more aware of their actual surroundings, prioritizing safety in more compact play spaces. This will mitigate many of the common avenues for injury that can occur when deeply immersed in VR. This allows the player enjoy a safer and more immersive VR experience.

Acknowledgements

We would like to give thanks to Professor Tyler Bell for use of his lab and equipment, Brian Busch for the use of the machine shop, and Matthew Miller for hardware design considerations and laser-related activities.

Final Prototype



In-Game Usage