A young man with short dark hair, wearing a blue t-shirt with the IEEE logo, is focused on working with a piece of cardboard. He is holding the cardboard with both hands, and a yellow piece is visible on the surface. The background is a blurred workshop or classroom setting with other people and a laptop. A semi-transparent dark grey box is overlaid on the image, containing white text.

# Daydreaming in Class: Building Virtual Reality Apps on Mobile

An IEEE Computer Society Workshop



**SEAN KIRMANI**

- EE and CS major
- Virtual and Augmented Reality Intern @ Google
- IEEE CS Officer
- Wants To Help You Learn About VR and AR

# What This Workshop is Supposed To Do

Let you try virtual reality, if you haven't.

Get you started on making a VR app on your phone.

Introduce you to some VR and AR design concepts.

Most of the time will be spent on working on your apps!

We're here to answer some of your questions and help you!

# The State of Virtual and Augmented Reality

- High end VR headsets: Facebook's Oculus Rift, HTC Vive
- Cheap and accessible VR: Google Cardboard, Oculus Gear VR
- Prototypical AR Headsets: Microsoft HoloLens, Meta AR Headset
- AR in Your Pocket: Google's Project Tango

# Introduction to Virtual Reality Design

# A new dimension

Designing great user experiences in virtual reality is considerably different than designing for traditional 2D form factors. Informed by fundamental aspects of human perception and cognition, they will help you design applications for virtual reality on any platform.

This is an exciting era for interactive, visual, and motion graphics designers.

# Avoiding simulator sickness

Virtual reality introduces a new set of physiological considerations for design. Like flight simulators used by pilots in training, virtual reality has the potential to present mismatches between physical and visual motion cues. This mismatch can produce nausea known as “simulator sickness,” when your eyes think you’re moving, but your body does not.

# Establishing familiarity

As a new medium, users may not be familiar with virtual reality interactions yet. Because the virtual reality canvas is infinite, guiding user focus and attention presents unique challenges not present in 2D applications.



# Physiological considerations

# Psychological considerations

Motion sickness is caused by the disparity between what one feels and what one expects to feel.

Evolution gives clues about why this disparity exists. The ability to feel motion sickness evolved as a survival skill.

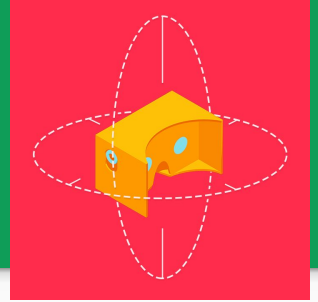
In the modern age, motion sickness is more of a nuisance than a benefit.

# Head tracking

Head tracking enables objects in virtual space to maintain fixed positions regardless of how you move your head, creating the perception of a virtual world surrounding you.

The most important guideline in designing for virtual reality is to always maintain head tracking. Never stop tracking the user's head position inside of the application. Even a short pause in head tracking will cause some users to feel ill.

# Head tracking



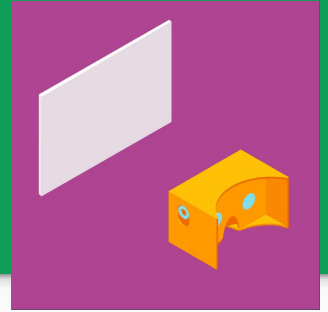
Render 2D splash screens in 3D space

Avoid slight freezes in head tracking

What if you can't have head tracking turned on?

- Maintaining audio feedback when the screen fades to black will help establish that the app is still running, and indicate it is loading new information.
- Filling the entire screen with a changing color or brightness level can provide visual feedback without the disorientation of turning off tracking.

# User control of movement



It is important that users are not “passengers” in the app as much as active drivers. Users should remain in control of their movement inside the app, so that they will be able to actively anticipate what they are about to see.

Fundamentally, users should still remain in control. It is better to let them start the roller coaster, versus having it start on its own.

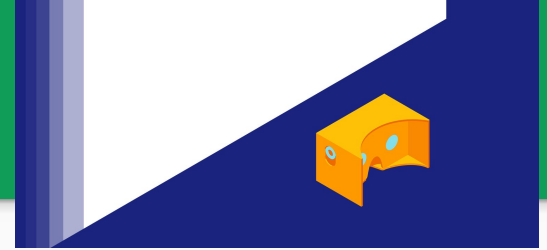
# Use constant velocity

In real life, we feel acceleration and deceleration, but we do not feel velocity. For example, when flying in an airplane, we feel the takeoff and landing.

When the user virtually accelerates or decelerates inside of your application, they will not feel the change in real life. The disparity between what they are seeing and what they are feeling may cause discomfort.

You can reduce this discomfort by trying to keep the user at a constant velocity when they are moving inside of your app.

# Grounding with fixed objects



If the user is sitting while using a virtual reality device, placing the user in a stationary virtual environment (such as in a virtual chair) will help them reconcile the sensation of movement in VR with the fact that they are not moving in real life.

If the user is near a very large virtual object that is moving, they may mistakenly believe that they are moving, as opposed to the large object moving.

This may cause discomfort. Avoid this by including more fixed points of reference in the user's environment.

# Brightness changes

Be mindful of sudden changes in brightness.

Given the proximity of the screen to the user's eyes, transitioning the user from a dark scene to a bright scene may cause discomfort as they acclimate to the new level of brightness.

It is similar to stepping out of a dark room into the sun.



# Interactive patterns

# Controls

When your app launches, place the UI controls in the user's current field of view. If the controls are placed out of view, such as directly north, users will often wait, feel confused, and have to look around for the controls.

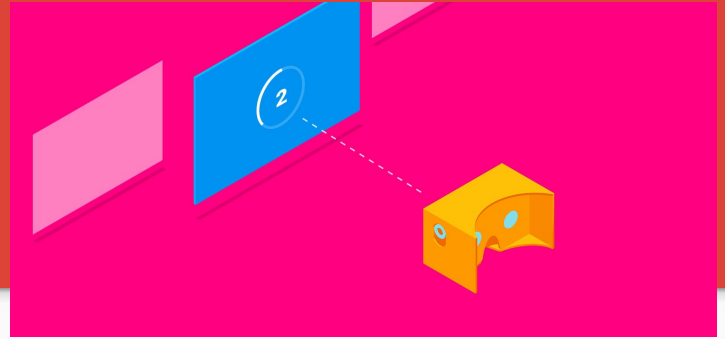
If your application allows movement, update the location of the UI controls based on the user's changing position and field of view. Without updating the location, the user will have to travel back to their original position to find them.

# Fuse buttons

Devices like Cardboard have a button on the side of the device for clicking on targets. An alternative to using this button is to create virtual buttons that leverage a timer, or fuse, which trigger after the user has focused on them for a certain amount of time.

These buttons are becoming increasingly common in virtual reality apps; however, they can feel slow and frustrating as the user must wait for the button to activate. When creating fuse buttons, always give the user the option to directly and immediately click on the target.

# Visual countdown



When using a fuse button, visually represent the countdown to activation so that the user knows what is happening. Failing to show the countdown can easily cause misclicks, and the user may not understand why the app suddenly changed state when they were just looking around.

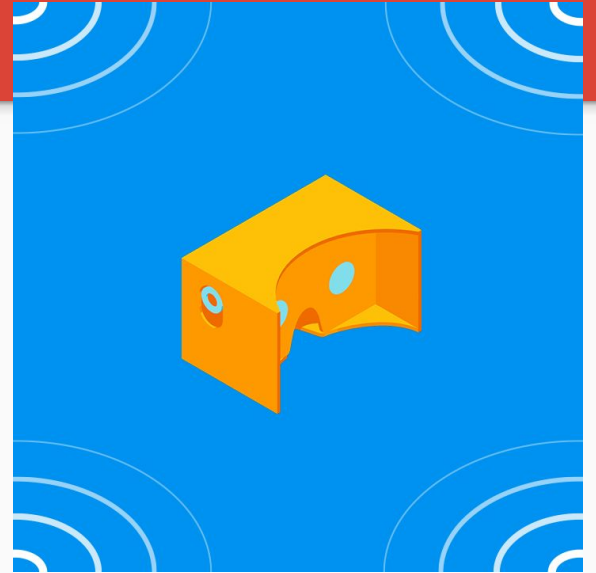
# Button placement

Avoid placing fuse buttons in close proximity to each other. Fuse buttons work best if they are large targets that are sufficiently far apart from each other.

If multiple smaller fuse buttons are placed near each other, the user could accidentally click on the wrong button. Smaller buttons that are close to each other should require a direct click to activate.

# Feedback

- Audio
- Haptic feedback



# Audio

Text instructions don't perform well in virtual reality for a number of reasons. Small text is hard to read. Users are often overloaded with visual information from the virtual environment around them. Similar to textual instructions, brevity is important.

## **Audio as background**

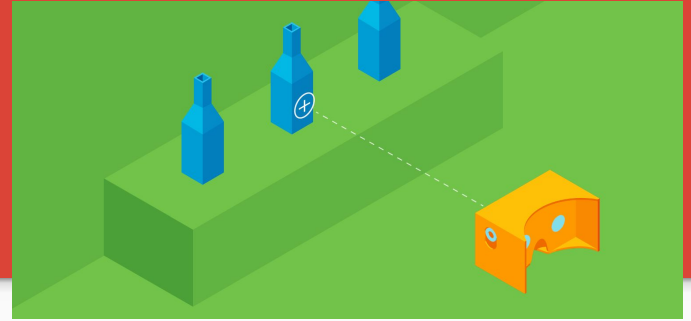
Virtual reality applications are meant to be immersive. Users regularly wear headphones. Consider using environmental audio to make the application more realistic, and to draw the user's attention to various areas of the app.

# Haptic feedback

Haptic feedback – information from the sense of touch – improves the user's tactile experience in VR. Events such as a user touching an object or interacting with controls can benefit from it. Even though haptic feedback isn't often high fidelity on a phone handset, it makes the experience more tangible.



# Display reticle



Users can somewhat accurately target small objects just by moving their gaze. However, it is considerably easier for users to do fine targeting tasks if given a visual reticle (visual aid to track targets) to help them aim. Display a reticle when the user is doing fine targeting.

## Reticle size

If you display a reticle, render it stereoscopically (creating the illusion of depth) and so that it projects spatially onto targeted objects.

# Other VR Design Problems

Depth cues matter.

Changes in frame rate can make you sick.

Positional tracking is hard and can be dangerous.

**Design problems are in infancy stage.**

What About Augmented Reality?

# Designing for AR

- Real world spatial app design is a new challenge.
- Interacting with virtual objects in the real world.
- Making AR objects looking photorealistic.
- How do we deal with occlusion?

Still a lot of technical and design problems, so this is a younger problem...



**That's all! Get  
started developing!**