Web VR

COMP 531
Chongzhang “Christopher” Li
cl46@rice.edu

Outline

1. VR Basics
2. WebVR Overview
3. WebVR code demo

VR BASICS

What is VR?

Hacking user's perception system to fully **immerse** the user into a virtual experience.

Visual Immersion

+

Credit: https://people-mozilla.org/~vladimir/misc/webvr-fitc.pdf

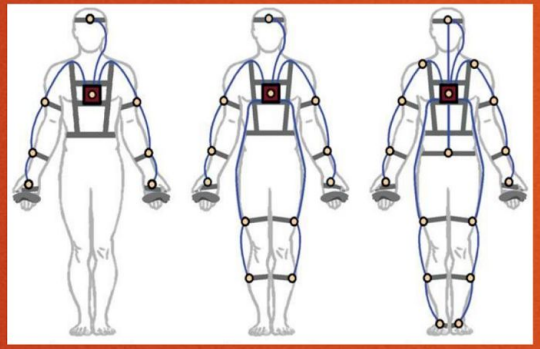
Per-Eye Rendering

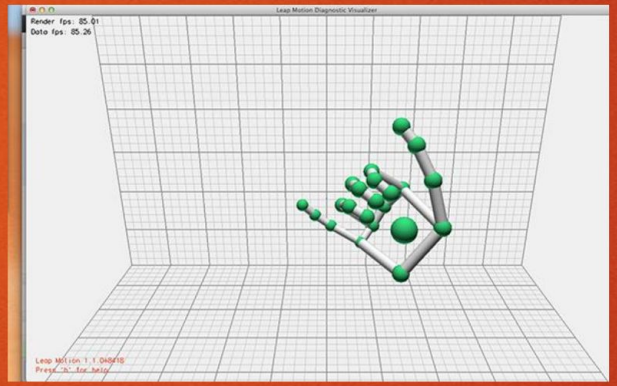
Credit: https://people-mozilla.org/~vladimir/misc/webvr-fitc.pdf

Device-Specific Distortion

Credit: https://people-mozilla.org/~vladimir/misc/webvr-fitc.pdf

Input Immersion





Credit: https://people-mozilla.org/~vladimir/misc/webvr-fitc.pdf

WebVR Overview

What is WebVR?

WebVR is a JavaScript API for creating immersive 3D, Virtual Reality experiences in your browser.

Why WebVR?

Pros:

- Hardware Agnostic: Same code easily accessible across VR devices
- Easy for Users: Does not require downloading and installing specialized software per experience
- Open, Accessible, and Linked

Cons:

- Lag: Performance may be worse than native application
- Development Stack: Forced to use Javascript and WebGL
- Under Development: Not yet supported by all browsers; API subject to change

Development for WebVR

Get access to the desplay

```
_getDisplays () {
   return navigator.getVRDisplays().then(displays => {
     // Filter down to devices that can present.
     displays = displays.filter(display => display.capabilities.canPresent);
     // Store the first display we find. A more production-ready version
should
     this._vr.display = displays[0];
     this._vr.display.depthNear = DemoVR.CAMERA_SETTINGS.near;
     this._vr.display.depthFar = DemoVR.CAMERA_SETTINGS.far;
   });
 }
```

Render the scene for each eye

```
// Get all the latest data from the VR headset and dump it into
frameData.
   this._vr.display.getFrameData(this._vr.frameData);


// Left eye.
   this._renderEye(
     this._vr.frameData.leftViewMatrix,
     this._vr.frameData.leftProjectionMatrix,
     {
       x: 0,
       y: 0,
       w: EYE_WIDTH,
       h: EYE_HEIGHT
     });
```

```
// Right eye.
   this._renderEye(
     this._vr.frameData.rightViewMatrix,
     this._vr.frameData.rightProjectionMatrix, {
       x: EYE_WIDTH,
       y: 0,
       w: EYE_WIDTH,
       h: EYE_HEIGHT
     });
   // Use the VR display's in-built rAF
   this._vr.display.requestAnimationFrame(this._update);
   // Call submitFrame to ensure that the device renders the latest
image from the WebGL context.
   this._vr.display.submitFrame();
```

```
_renderEye (viewMatrix, projectionMatrix, viewport) {
   // Set the left or right eye half.
   this._renderer.setViewport(viewport.x, viewport.y, viewport.w,
viewport.h);


   // Update the scene and camera matrices.
   this._camera.projectionMatrix.fromArray(projectionMatrix);
   this._scene.matrix.fromArray(viewMatrix);


   // Tell the scene to update (otherwise it will ignore the change of
matrix).
   this._scene.updateMatrixWorld(true);
   this._renderer.render(this._scene, this._camera);
 }
```

Recap

1. What is VR?
2. Visual Immersion
3. WebVR pros and Cons
4. Development with WebVR

Reference

- https://people-mozilla.org/~vladimir/misc/webvr-fitc.pdf
- http://web.mit.edu/hawksley/Public/IntroToWebVR/Intro-WebVR-slides.pdf
- https://developers.google.com/web/fundamentals/vr/getting-started-with-webvr/