WebVR with three.JS

Background

- Enable VR in Web platform
- Webpage
  - https://webvr.info/

- Platform
  - Chromium (experimental Chrome)
  - Firefox Nightly (experimental Firefox)
  - Samsung Internet Browser for GearVR

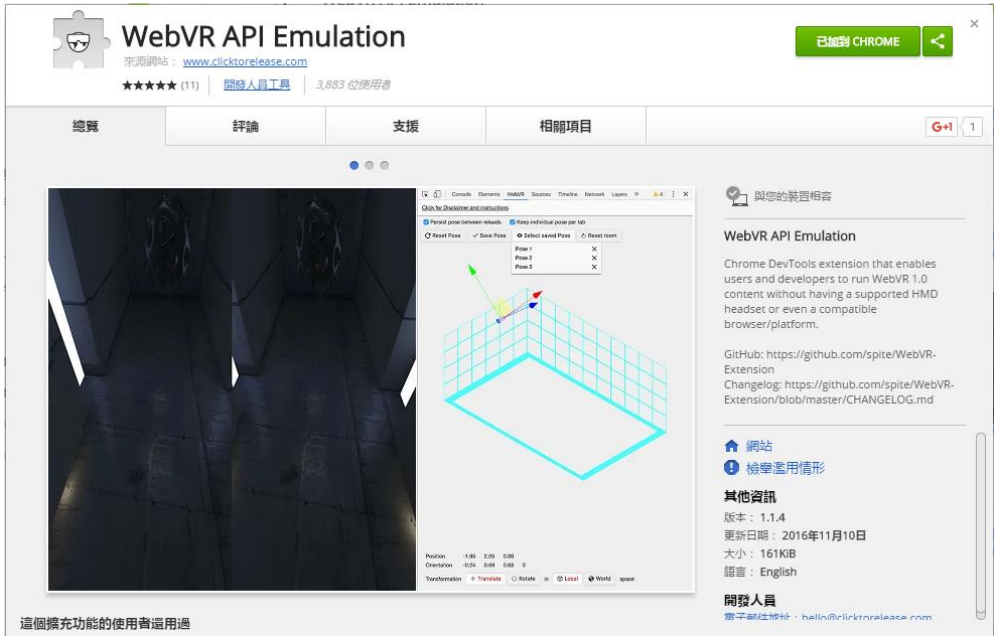
News

- Chrome is starting to support for WebVR
  - From Feb 2017
  - For **Android only**

- Reference:
  - http://marketingland.com/google-releases-chrome-version-first-browser-support-webvr-206314

Installation on Windows

- Method 1
  - Download WebVR-enabled browser
- Method 2 (Run without HMD)
  - Chrome WebVR Extension



Simple HTML Server

- Some machines cannot run WebVR ***locally***

- Solution: Run a local html server
  - Install Python
    - https://www.python.org/
  - Run HTML server with cmd on your webpage directory
    - *python -m SimpleHTTPServer 8000* (before 3.0)
    - *python -m http.server 8000* (3.0+)
  - Access the web page
    - http://localhost:8000/<yourfilename>

Start WebVR programing

Need to learn WebGL first

- Read our tutorial2-5

Tutorial 2

Attached Files: Tut2_BasicWebGL.pdf (784.226 KB)
tut2.zip (4.573 KB)

Tutorial 3

Attached Files: Tut3_Threejs.pdf (419.116 KB)
tut3.zip (257.387 KB)

Tutorial 4

Attached Files: Tut4_Threejs_lightTex.pdf (448.126 KB)
tut4.zip (640.044 KB)

Tutorial 5

Attached Files: Tut5_Threejs_ObjMouse.pdf (454.388 KB)
tut5.zip (1.469 MB)

Now

- We use tut5 as an example
  - File: webgl_loader_obj.html
  - Add VR content into it

Include WebVR

- Files needed in three.js:
  - three.js/examples/js/vr/WebVR.js
  - three.js/examples/js/effect/VREffect.js
  - three.js/examples/js/controls/VRControls.js

- To include files in JavaScript

```
<script src="./webvr/WebVR.js"></script>
<script src="./webvr/VREffect.js"></script>
<script src="./webvr/VRControls.js"></script>
```

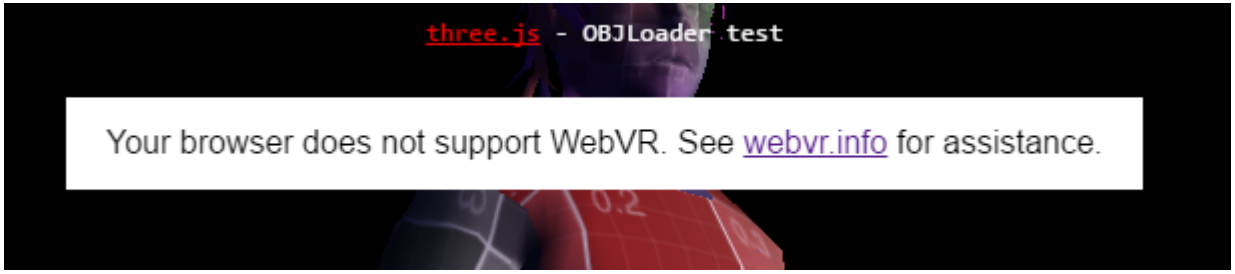
Check WebVR

- Check whether webVR is supported
  - Add the code in init()

```
function init() {

    if ( WEBVR.isAvailable() === false ) {
        document.body.appendChild( WEBVR.getMessage() );
    }
```

  - If webVR is not supported



Init WebVR

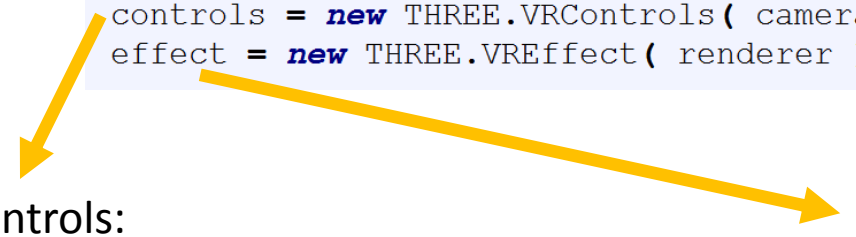
- Add new global variables

```
var effect,controls;
```

- New the VR objects in init()

```
renderer = new THREE.WebGLRenderer();
renderer.setSize( window.innerWidth, window.innerHeight );
container.appendChild( renderer.domElement );


controls = new THREE.VRControls( camera );
effect = new THREE.VREffect( renderer );
```



VRControls:
For VR Input

VREffect:
For VR Output

Render Stereo Effect

- Use the *effect* object similar to WebGL renderer
  - Change it with the new object

Change to

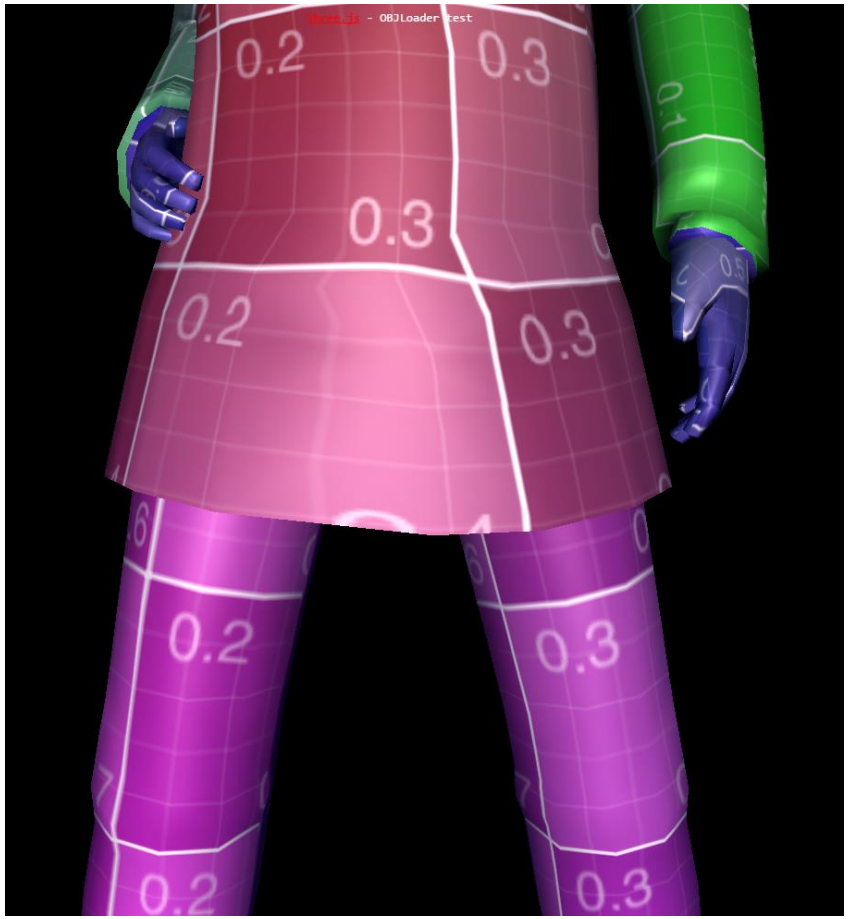
```
//renderer.setSize( window.innerWidth, window.innerHeight );
effect.setSize( window.innerWidth, window.innerHeight );
```

Change to

```
//renderer.render( scene, camera );
effect.render( scene, camera );
```

Result

- No stereo effect?
- WebVR allows us to on/off VR effect
  - Off by default



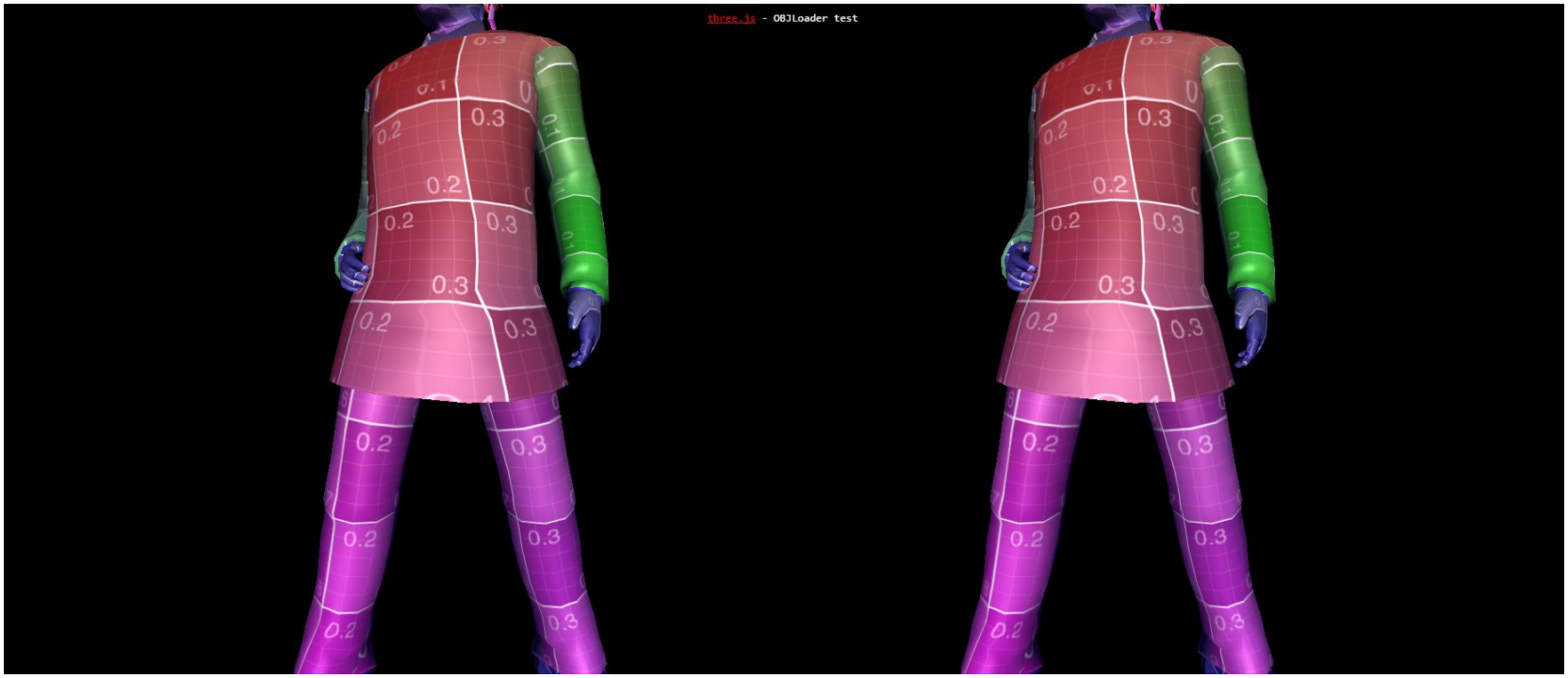
Enable VR Effect

- Adding a button at the bottom of screen

```
document.body.appendChild( WEBVR.getButton( effect ) );
```

- Click this button to enable VR

Result



Head Tracking

- Remember we have the VR input/output objects

```
controls = new THREE.VRControls( camera );
effect = new THREE.VREffect( renderer );
```

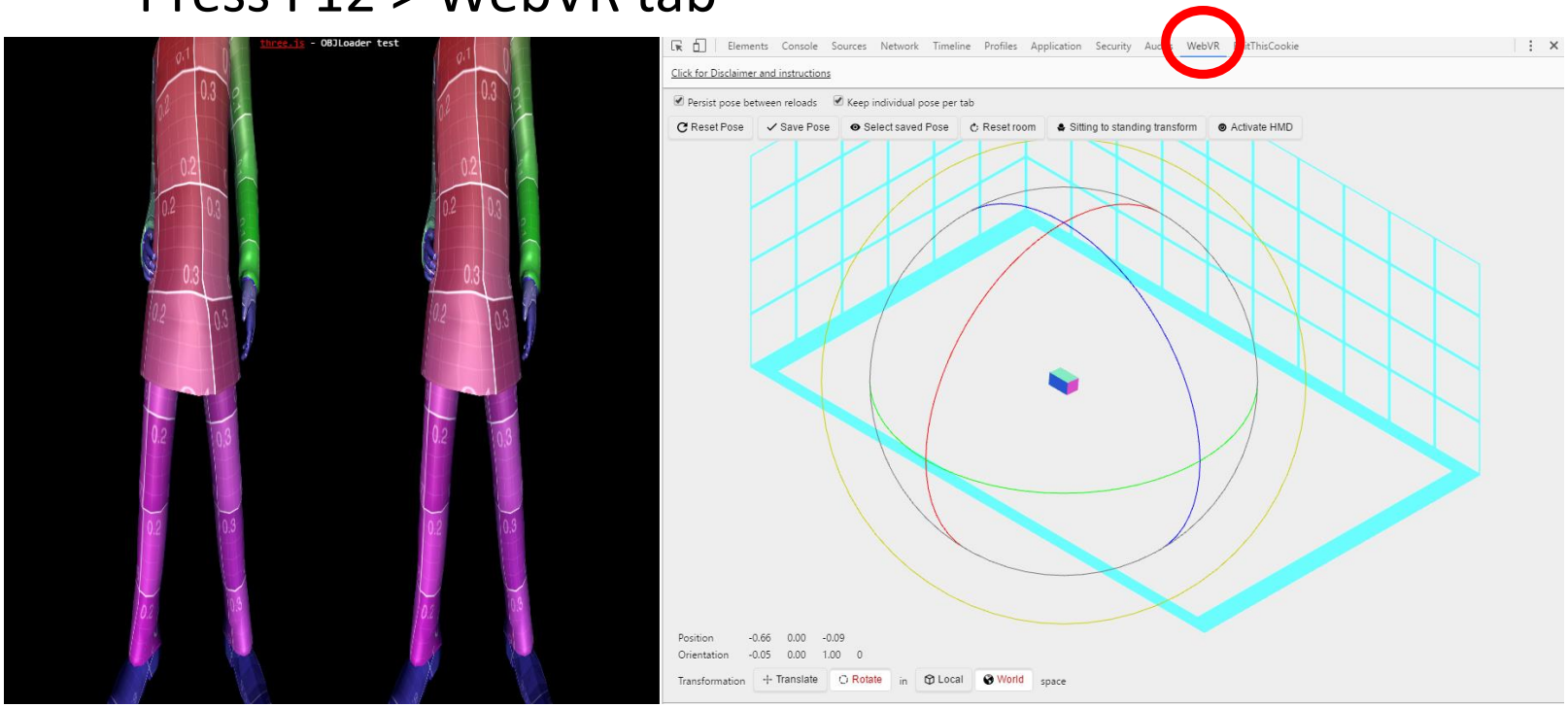
- To render with the latest head position/orientation

```
function render() {
    controls.update();
    effect.render( scene, camera );
}
```

  - Call the *update()* before *render()*

Testing the Tracking

- For Chrome extension, you can simulate the position/orientation of head in Chrome Developer Tools
  - Press F12 > WebVR tab



Run It on Real Device

Requirements

- WebVR-enabled browser
  - Chromium
  - FireFox Nightly

- All experimental browsers
  - Error prone

Chromium

- https://webvr.rocks/chromium

Instructions

1. Download the experimental build of Chromium with WebVR support.
2. Find the `chromium_webvr_v1.1_win.zip` file in your `Downloads` directory, and extract all the contents.
3. In the output directory, find and launch the `chrome.exe` file.
4. In the URL bar, load `chrome://flags#enable-webvr` and toggle the `Enable WebVR flag`.
5. In the URL bar, load `chrome://flags#enable-gamepad-extensions` and toggle the `Enable Gamepad Extensions` flag.
6. Launch the SteamVR application.
7. Enjoy WebVR content!

Nightly

- https://webvr.rocks/firefox

Setup Instructions

1. Download and launch Firefox Nightly.
2. Download version 1.0.2 (64-bit) of the `openvr_api.dll` file from the OpenVR GitHub repository.
3. Save the `openvr_api.dll` file somewhere on your computer where the user running Firefox can read it (e.g., `c:\openvr\`).
4. In Firefox Nightly, navigate to `about:config`; change the value of `dom.vr.openvr.enabled` to `true` and `gfx.vr.openvr-runtime` to the full path of the `openvr_api.dll` file (e.g., `c:\openvr\openvr_api.dll`).
5. Ensure that your Oculus Home settings allow for Unknown Sources.
6. Restart Firefox Nightly.
7. Enjoy WebVR content!