

Cartographie Réalité augmentée Geolocalisation et capteurs

Christophe Vestri

Le mardi 6 mars 2018

Objectifs du cours

- Bases de Cartographie et géographie
- Outils de Cartographie, Géographie en Html5 et JS
- Expérimenter quelques méthodes et outils
- un peu de VR
- Réaliser un projet en RA/Carto
- Evaluation:
 - Présence (20%)
 - Participation en classe (40%)
 - Projet (40%)

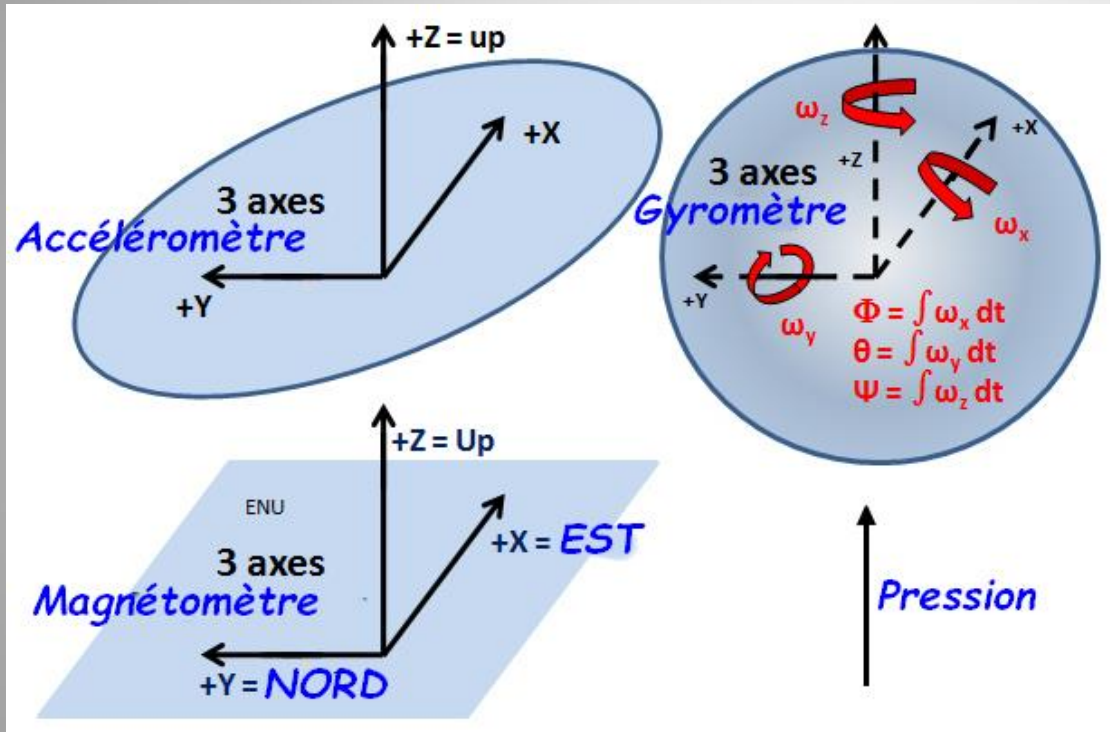
Plan du cours

- 20 février : Intro Carto/géo Leaflet
- 6 mars: Capteur et Geoloc/access en JS et Unity
- 13 mars: WebRTC, WebGL et Three.js
- 20 mars: Aframe AR.js et VR
- 27 mars : MapBox et Projets

Plan Cours 2

- Rappel dernier cours
- WebGL
 - Basics
 - GLSL
- Three.js
 - Device Events
 - Exercices
- Projet final

Capteurs smartphones



C'est donc un système à 10 capteurs d'attitude qui est embarqué

= 3 accéléromètres
+ 3 gyromètres
+ 3 magnétomètres
+ 1 pression

Html5

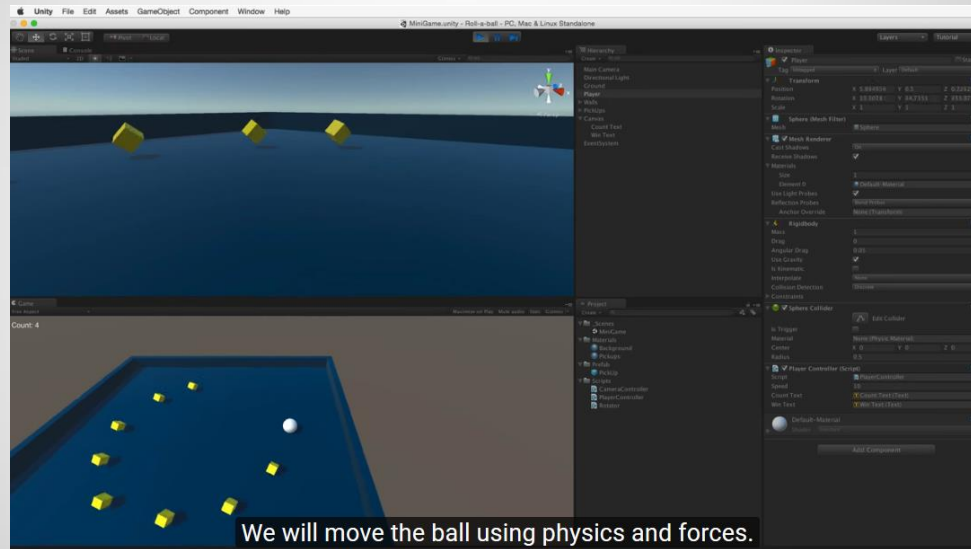
- **Acces capteur caméra:**
- **DeviceOrientation, DeviceMotion**
- **Caméra, Audio, Géolocalisation**
- **touchevents/mouse/...**
- **<https://developers.google.com/web/fundamentals/native-hardware/device-orientation/>**

Pour tester sur un Mobile

- Créer un compte sur <https://www.000webhost.com/>
- Ou tout autre free webhosting site
- Uploader vos fichiers
- Tester avec votre smartphone

Roll-a-ball with Gyromètre

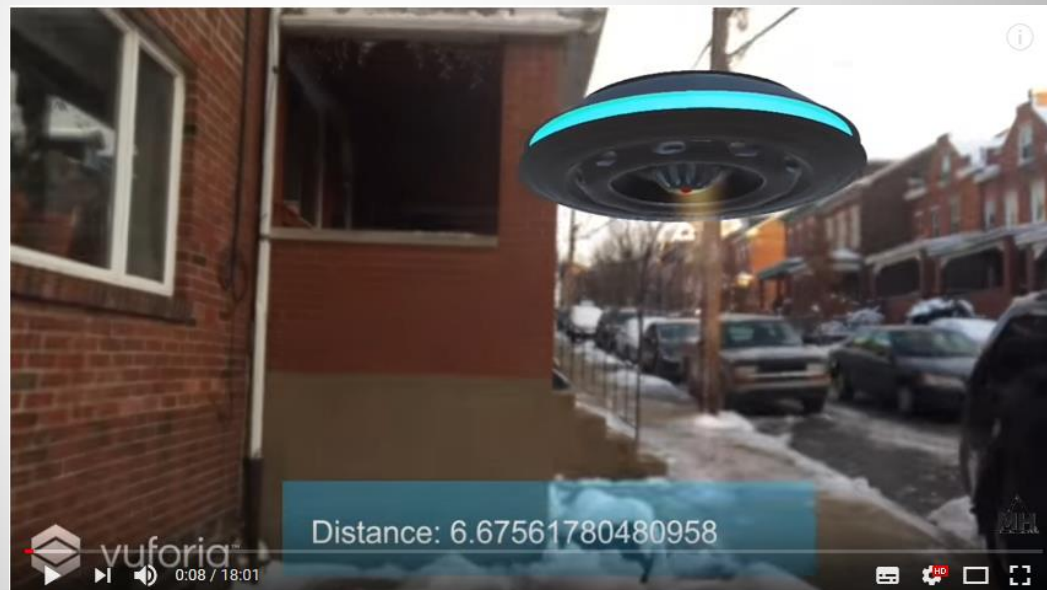
- <https://unity3d.com/fr/learn/tutorials/s/roll-ball-tutorial>



- Ajouter ensuite Gyromètre
- <https://docs.unity3d.com/ScriptReference/Input.html>

Geolocalisation Avec Unity

- <https://docs.unity3d.com/ScriptReference/LocationService.html>
- <https://docs.unity3d.com/ScriptReference/Screen-orientation.html>



- **Matthew Hallberg: Markerless AR**
- <https://www.youtube.com/watch?v=X6djed8e4n0>

WebRTC



- **Qu'est-ce que c'est?**
 - **Real-Time Communications (RTC) à travers une simple API**
 - **3 taches:**
 - **Acquisition audio et video (Mediastream)**
 - **Communication audio et video (RTCPeerConnection)**
 - **Communication d'autres données (RTCDataChannel)**
 - **<https://webrtc.org/>**

Example

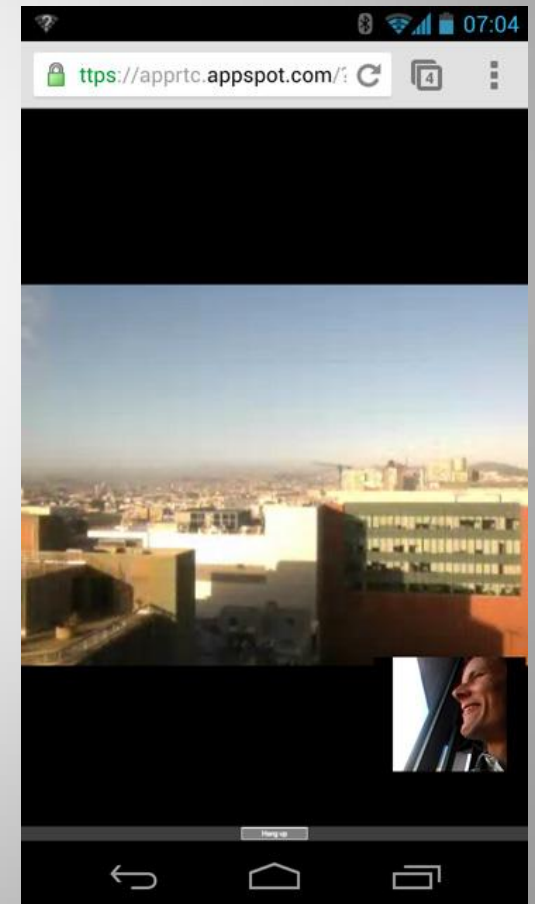
- <https://github.com/webRTC/samples/blob/gh-pages/src/content/getusermedia/audio/js/main.js>

```
11 // Put variables in global scope to make them available to the browser console.
12 var audio = document.querySelector('audio');
13
14 var constraints = window.constraints = {
15     audio: true,
16     video: false
17 };
18
19 function handleSuccess(stream) {
20     var audioTracks = stream.getAudioTracks();
21     console.log('Got stream with constraints:', constraints);
22     console.log('Using audio device: ' + audioTracks[0].label);
23     stream.oninactive = function() {
24         console.log('Stream ended');
25     };
26     window.stream = stream; // make variable available to browser console
27     audio.srcObject = stream;
28 }
29
30 function handleError(error) {
31     console.log('navigator.getUserMedia error: ', error);
32 }
33
34 navigator.mediaDevices.getUserMedia(constraints).
35     then(handleSuccess).catch(handleError);
```

WebRTC



- Chrome, Chrome for Android
- Firefox
- Opera



WebGL



- **Qu'est-ce que WebGL**
 - Cross plateforme et libre de droits
 - OpenGL ES (OpenGL simplifié pour l'embarqué) dans le Web (HTML5)
 - Bonne intégration Html et mécanisme d'évènements
 - DOM API pour affichage 2D et 3D
 - Langage de type script (pas de compilation)
 - Accélérations matérielles et GPU

WebGL

WebGL - 3D Canvas graphics 📄 - OTHER

Usage % of all users

Global 93.75%

Method of generating dynamic 3D graphics using JavaScript, accelerated through hardware

Current aligned	Usage relative	Date relative	Show all							
IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome for Android	UC Browser for Android	Samsung Internet	
			49							
			63		10.2					
			64		10.3				4	
1 11	1 16	58	65	11	11.2	all	64	1 11.8	6.2	
	1 17	59	66	11.1	11.3					
		60	67	TP						
		61	68							

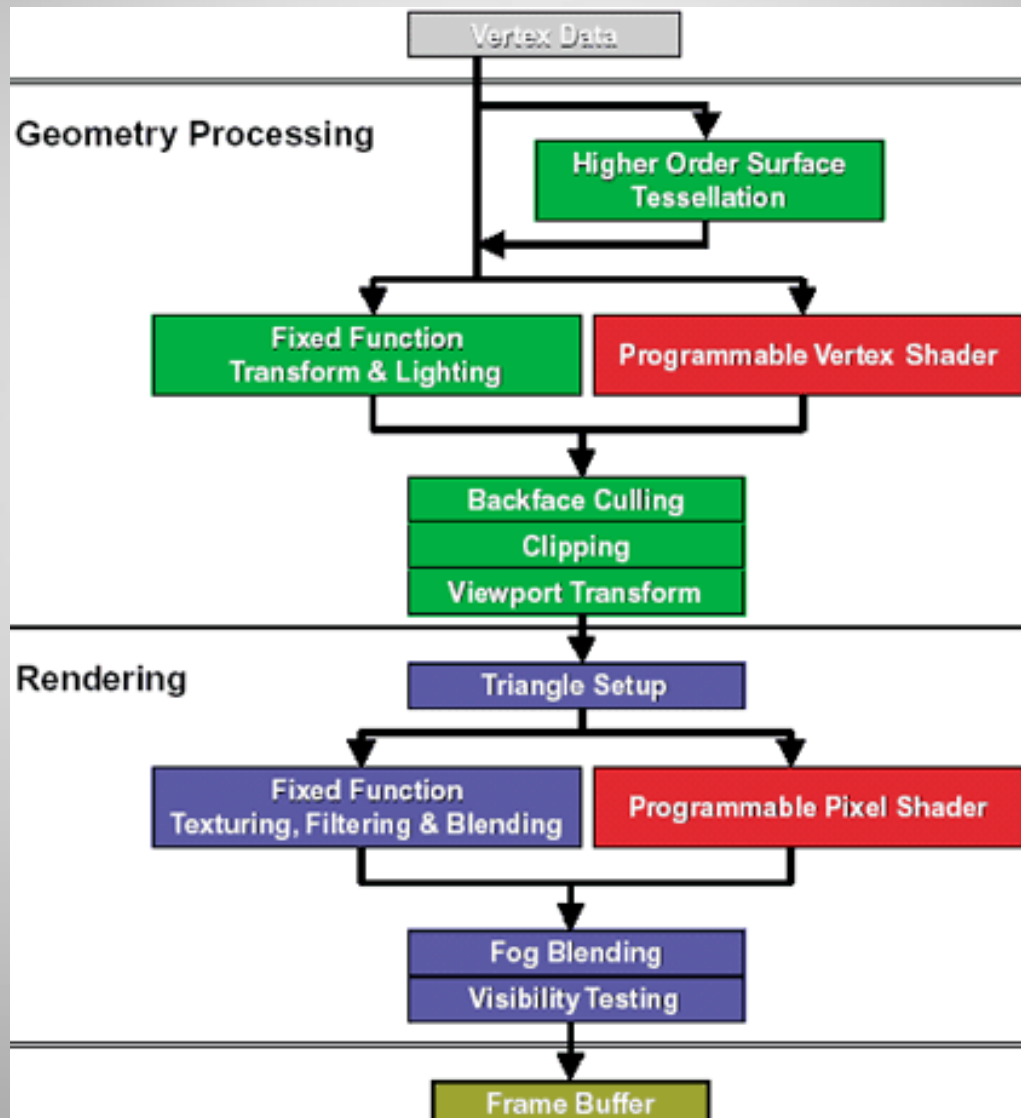
— Blacklist:

<https://www.khronos.org/webgl/wiki/BlacklistsAndWhitelists>

WebGL

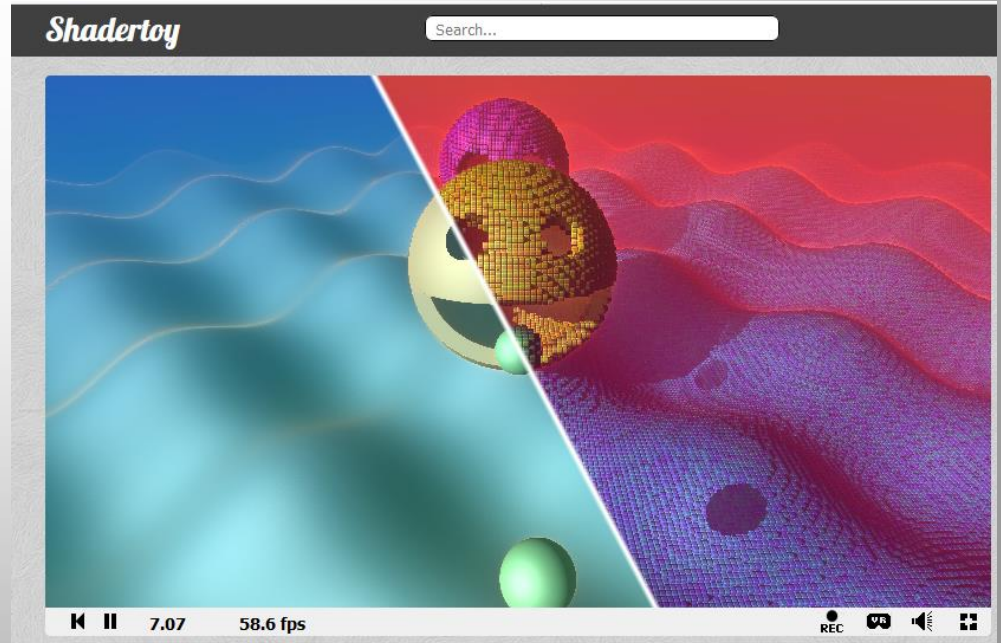
- **Low-level API**
 - GLSL OpenGL Shading Language
 - Machine d'état: OpenGL Context
 - Calcul de matrices et transformations
 - Buffers de vertex: positions, normals, color, texture
 - Depth buffer, Blending, transparency
 - Lighting, Cameras...
 - <http://www.webgltutorials.org/>
 - <https://www.khronos.org/webgl/wiki/Tutorial>

WebGL Pipeline



WebGL Examples

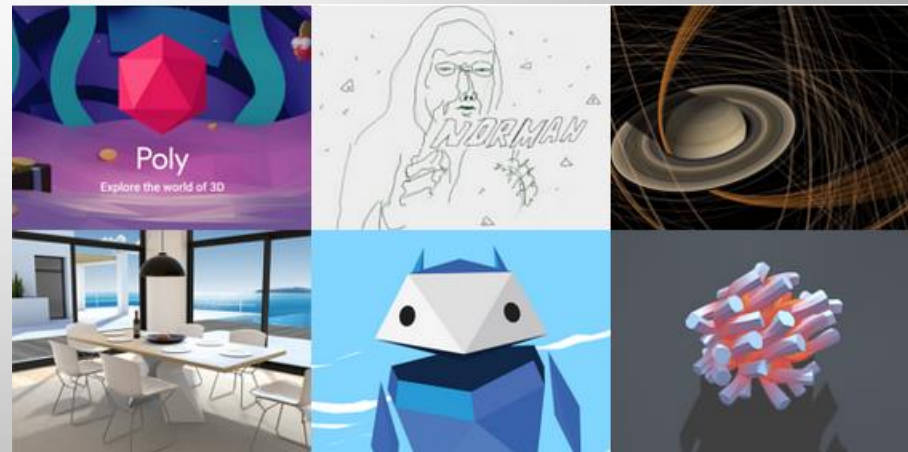
- https://developer.mozilla.org/fr/docs/Web/API/WebGL_API
- <https://webglfundamentals.org/>
- <https://www.shadertoy.com/>



Three.js

THREEJS

- Qu'est-ce que Three.js
 - Couche abstraite et haut niveau de WebGL
 - Librairie javascript pour créer des scènes 3D
 - Cross-plateforme et gratuit
 - Rendus en webGL, CSS3D et SVG
 - <https://threejs.org/>



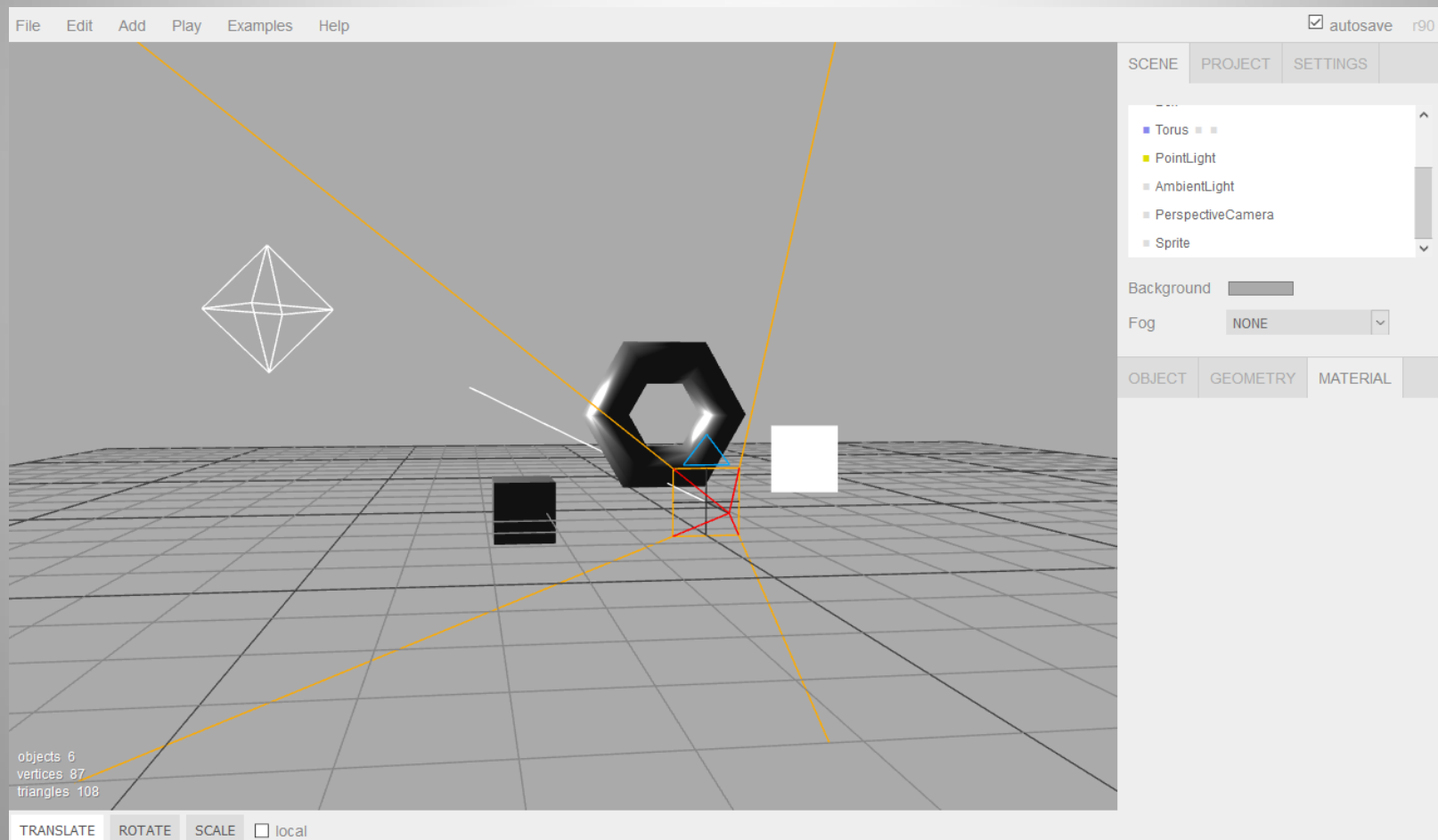
Fonctionnalités

THREEJS

- Scenes, Cameras, Renderer,
- Geometry, Materials, Textures
- Lights, Shadows
- Shaders, Particles, LOD
- Loaders: Json compatible Blender, 3D max, Wavefront OBJ, Autodesk FBX
- Animation, Trackballcontrols, Math Utilities

Threejs Editor

- <https://threejs.org/editor/>



Courses/Examples

- <http://davidscottlyons.com/threejs-intro/>
- <https://classroom.udacity.com/courses/cs291>
- <https://codepen.io/rachsmith/post/beginning-with-3d-webgl-pt-1-the-scene>
- <https://threejs.org/examples/>

Exercice

- Créez une scene + caméra + light + renderer
- Créez un objet
- Texturez cet objet
- Téléchargez un objet
- Animez l'objet (mouvement + déformation)
- Ajoutez Fog/pluie ou particules

Projet final

- **Projet final**
 - Un projet avec de la GéoLocalisation/capteurs
 - Un peu de RA si possible
 - Mélangez aruco/jsfeat/leaflet/geoloc/deviceApi
 - Afficher des objets Géolocalisés flottants, se balader sur une carte ou labyrinthe (Unity/js?)

Rappel

<https://github.com/artmobilis/ArtMobilis-js/wiki/fr-Configuration-framework-nodejs-ionic-android>

- **Chrome:**

- Bloque getUserMedia pour les fichiers locaux
- Lancer avec --disable-web-security pour du debug
- Navigator.getUserMedia plus supporté -> MediaDevices.getUserMedia()
- Il faudrait utiliser adapter.js
- Attention: exemples pas mis à jour -> utilisez Firefox

- **Firefox:**

- Version 40 et +: pb avec les vieilles cartes graphique blacklistées
- Installer version 31 pour du debug (marche sur mon laptop)