Teapots Can Fly

3D Graphics for Web Programmers

Presented by

Kelley Nielsen

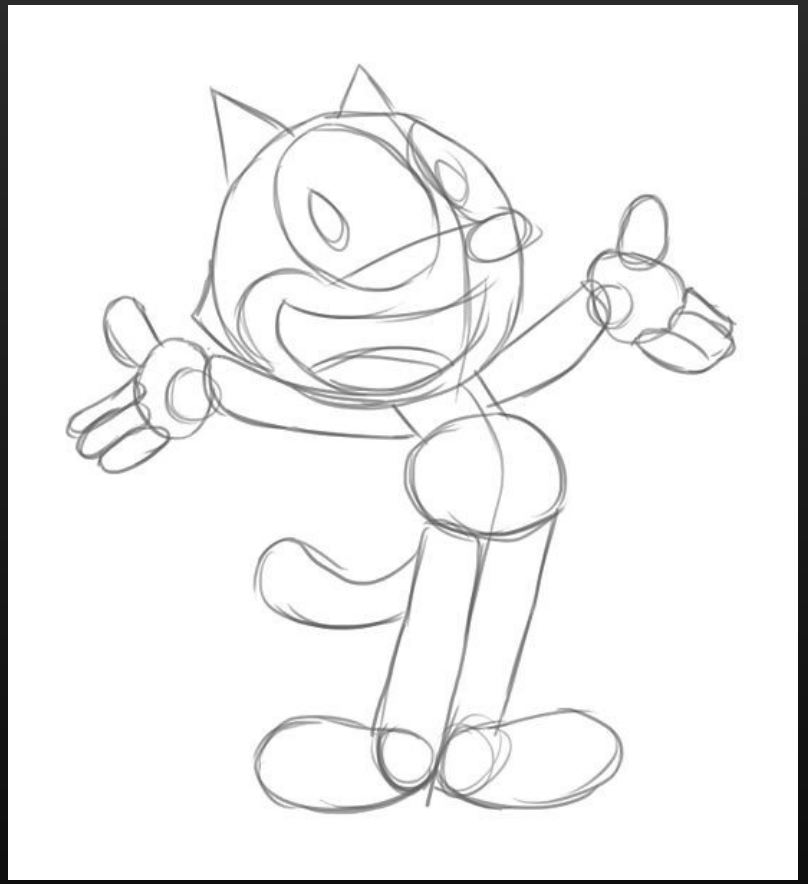
Salticid Software, Codechix

So,

how does

3D animation

work?

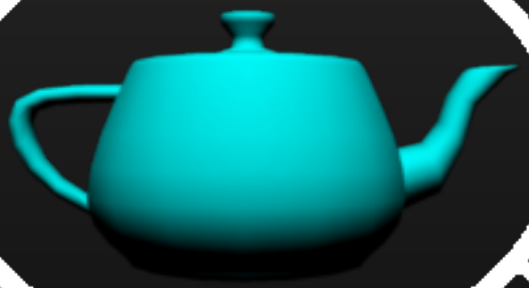


3D Animation is like Claymation

We're building

Our own little world

Where?



What?

How?

THREE.Scene

THREE.PerspectiveCamera

THREE.Mesh

THREE.PointLight

Affine transformations:
translate, rotate, scale

requestAnimationFrame()

How?

What do we need?

- three.js
- teapot.js
- An html5 page

To get it all:

https://github.com/shegeek/teapots_can_fly

clone or download zip

The basic setup:

```
<head>
  <title>Teapots can fly!<title>
  <style>canvas { width: 100%;
     Height: 100% }
  </style>
</head>
```

The basic setup:

```
<body>
  <script src="three.min.js">
  </script>
  <script>
     ** Our Stuff Goes Here! **
  </script>
</body>
```

The last setup step:

```
<script>

 var renderer = new THREE.WebGLRenderer();

 renderer.setSize(window.innerWidth,
                  window.innerHeight);

 document.body.appendChild
                    (renderer.domElement);

</script>
```

And now, the 3D code!

THREE.Scene

THREE.PerspectiveCamera

Our Diorama

```
var scene = new THREE.Scene();
```

Our Camera

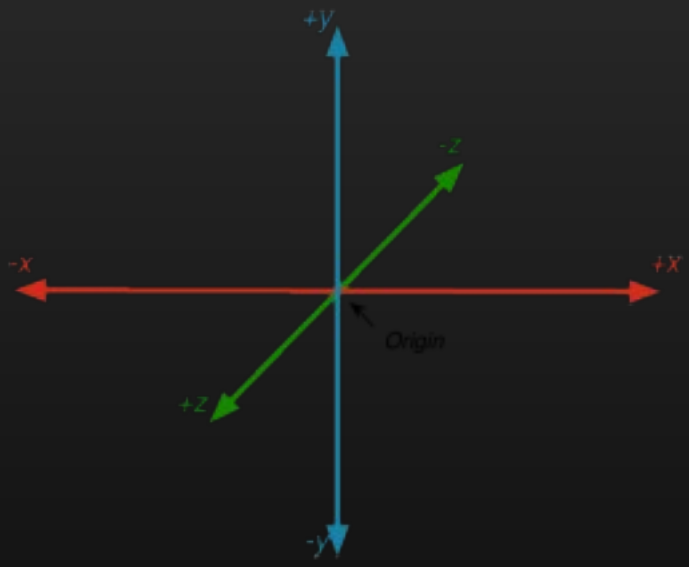
```
var camera = new THREE.PerspectiveCamera(
35,
window.innerWidth/window.innerHeight,
0.1, 1000);

camera.position.z = 50;
```

Right-handed coordinates

Positive X to the right

Positive Z coming out

of the screen



THREE.Mesh

THREE.PointLight

Our Teapot

```
var teapot;
var jsonLoader = new THREE.JSONLoader();
jsonLoader.load( "teapot.js", createTeapot);
```

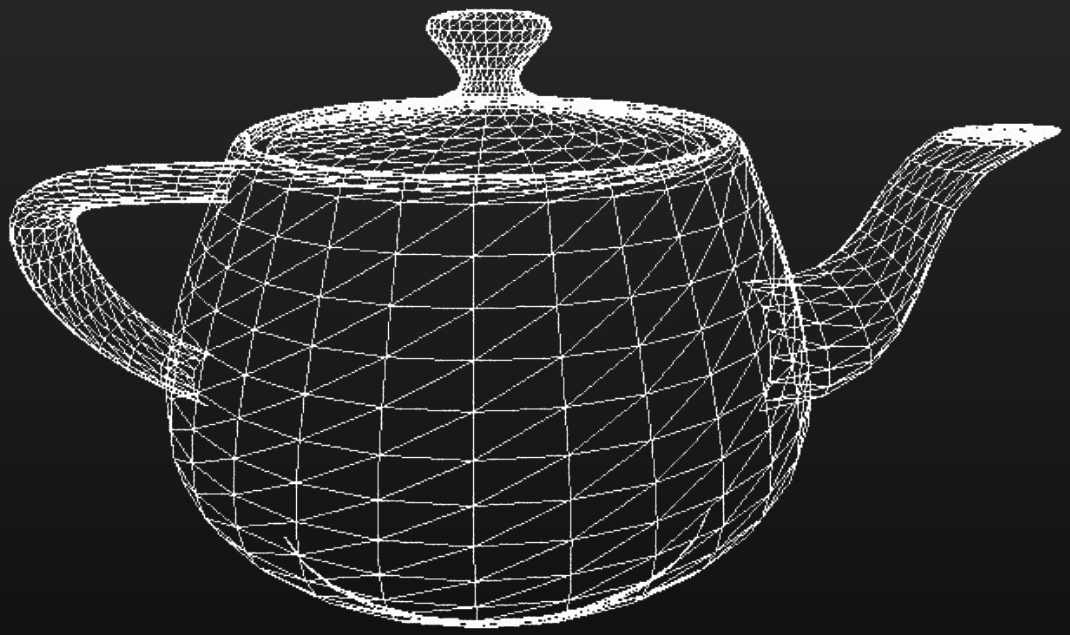
Our Teapot's Callback

```
function createTeapot(tGeometry){
  var tMaterial = new
        THREE.MeshPhongMaterial({color: 0x00ffff});
  var tMesh = new THREE.Mesh( tGeometry, tMaterial );
  scene.add(tMesh);
  teapot = tMesh;
}
```

A closer look...

```
new THREE.Mesh( tGeometry, tMaterial );
```

A mesh has two parts



A Geometry

Is like

Bones...

...A Material

Is like

Skin.

Let there be light!

```
var light = new THREE.PointLight(0xffffff);
light.position.set(0,150,150);
scene.add(light);
```

Affine transformations:
translate, rotate, scale

requestAnimationFrame()

How?

Our Render Loop

```
var halfScreenWidth = 80;
var render = function () {
  if (teapot) {
      teapot.position.x += 0.1;
      if (teapot.position.x > halfScreenWidth)
          teapot.position.x = -halfScreenWidth;
       renderer.render(scene, camera);
  }
requestAnimationFrame(render);
};
render();
```

Felix image courtesy of Wikihow
wikihow.com/Draw-Felix-the-Cat

Gumby image courtesy of
Art Clokey's Gumbyworld
gumbyworld.com

Earth image courtesy of NASA
visibleearth.nasa.gov

Coordinate axes image courtesy of
http://www.cocos2d-x.org/

Teapot wireframe image courtesy of
caig.cs.nctu.edu.tw/

Saran Wrap man image courtesy of
funnyordie.com

Resources and links

three.js repo: https://github.com/mrdoob/three.js
three.js home page: http://threejs.org/

Stemkoski's examples:
http://stemkoski.github.io/Three.js/index.html

WebGL Up and Running (by Tony Parisi):
http://shop.oreilly.com/product/0636920024729.do

Learning Three.js blog: http://learningthreejs.com/

Resources and links

three.js boilerplate builder:
http://jeromeetienne.github.io/threejsboilerplatebuilder/

An Introduction to Web GL:
http://dev.opera.com/articles/view/an-introduction-to-webgl/

Tutorials on the LearningWebGL blog:
http://learningwebgl.com/blog/?page_id=1217

WebGL 1.0 spec: http://www.khronos.org/webgl/

...And one again, the repo:

https://github.com/shegeek/teapots_can_fly

Enjoy! Make cool stuff!

kelleynnn@gmail.com