U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Interactive Graphics Systems

Pictures: two details of HCI from movie T*he Dark Knight (credits BLIND, Ltd.)*

Managing cameras

Alexandre Valle | DEI | FEUP

Requirements

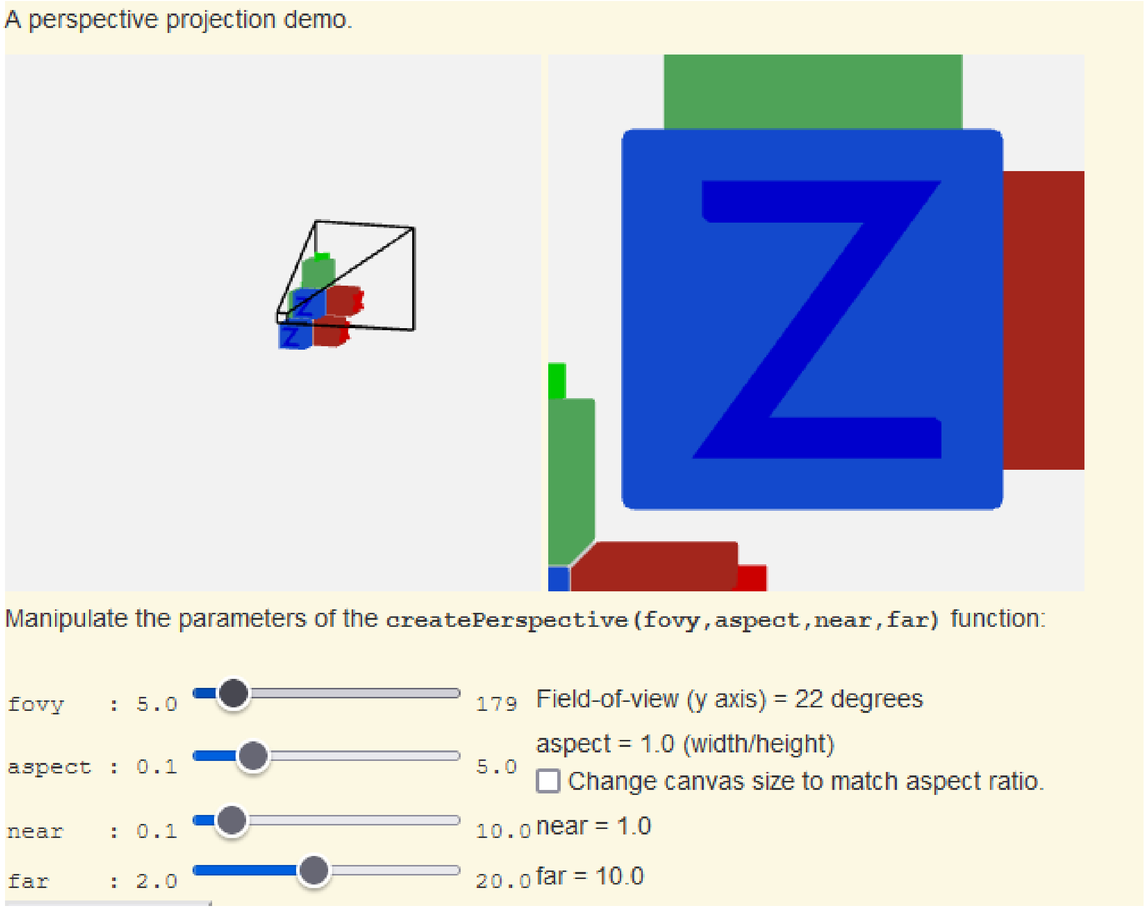
- WebGCF, which already contains CGFcamera and CGFcameraOrtho

CGFcamera by example: creation

```
// example of instancing a new perspective camera with a lens of 45º,
near and far clipping planes of 0.1 and 500 respective, located at
10,10,10 and pointing to the origin of axes

var c = new CGFcamera(45 * 3.14159 / 180, 0.1, 500,
vec3.fromValues (10,10,10), vec3.fromValues(0,0,0)
```

Illustration



A perspective projection demo.

Manipulate the parameters of the `createPerspective(fovy,aspect,near,far)` function:

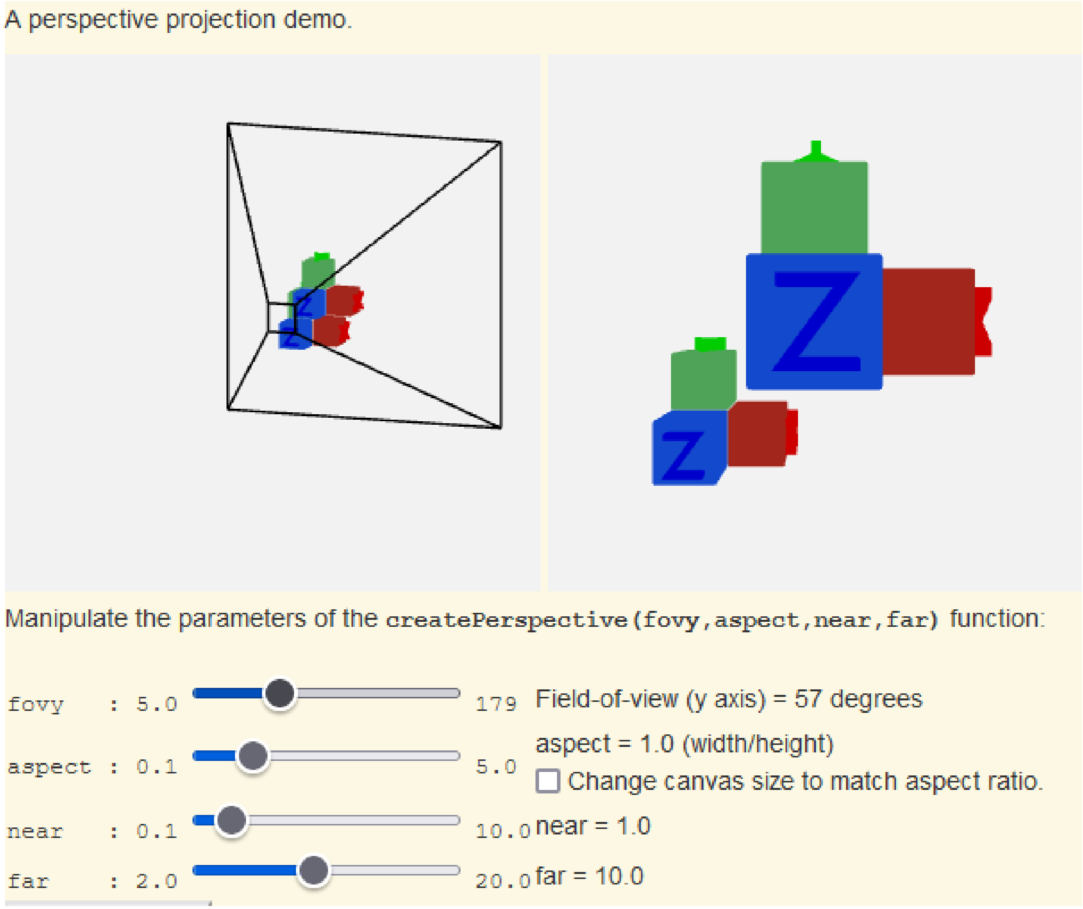
fovy : 5.0 179 Field-of-view (y axis) = 22 degrees

aspect : 0.1 5.0 aspect = 1.0 (width/height)

□ Change canvas size to match aspect ratio.

near : 0.1 10.0 near = 1.0

far : 2.0 20.0 far = 10.0



A perspective projection demo.

Manipulate the parameters of the `createPerspective(fovy,aspect,near,far)` function:

fovy : 5.0 179 Field-of-view (y axis) = 57 degrees

aspect : 0.1 5.0 aspect = 1.0 (width/height)

□ Change canvas size to match aspect ratio.

near : 0.1 10.0 near = 1.0

far : 2.0 20.0 far = 10.0

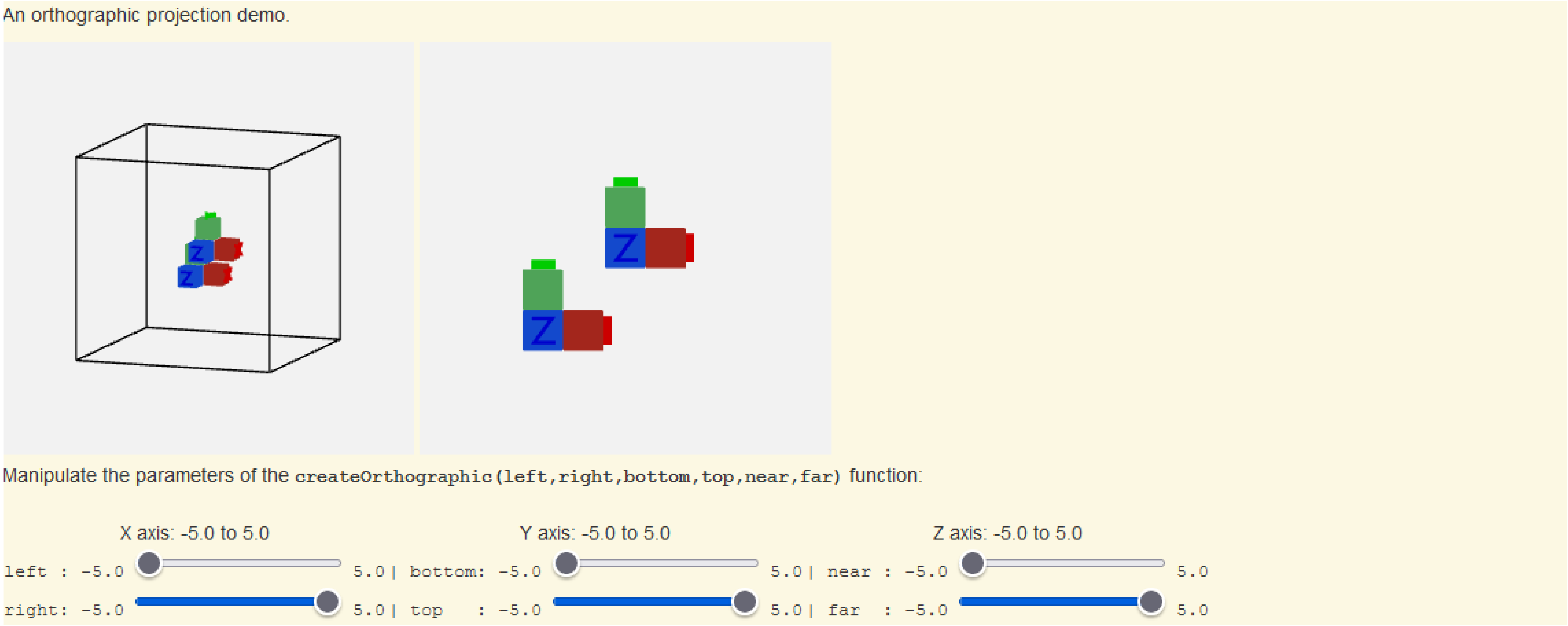
Source: https://learnwebgl.brown37.net/08_projections/projections_ortho.html

CGFcameraOrtho by exemple: creation

```
// example of instancing a new ortho camera for the cube -5 < x < 5, -5 < y < 5 and -5, 5 for near and far clipping planes respectivelly. Camera is located at 0,0,10 and pointing to the origin of axes. Up is aligned with the YY axis.

var c = new CGFcameraOrtho(-5, 5,-5, 5 -5, 5,
vec3.fromValues (10,10,10), vec3.fromValues(0,0,0),
vec3.fromValues(0,1,0)
```

Example 1/2



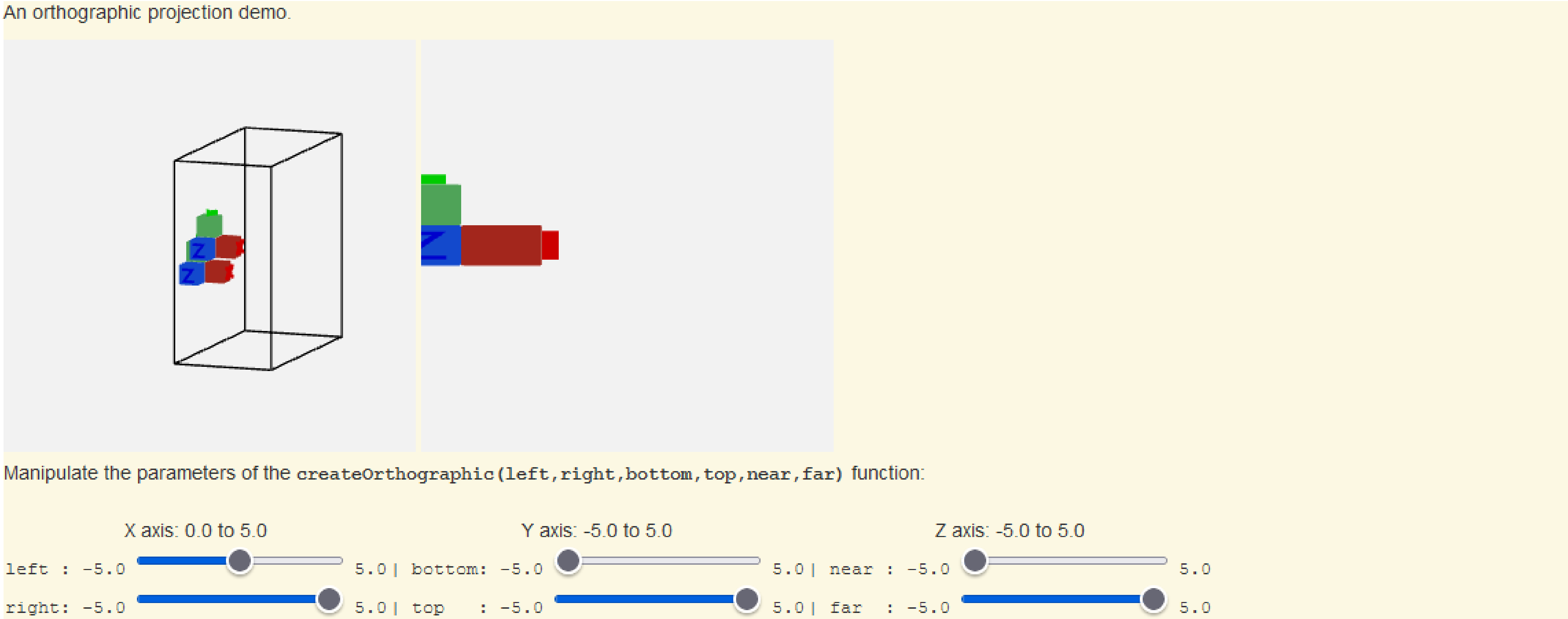
An orthographic projection demo.

Manipulate the parameters of the `createOrthographic(left,right,bottom,top,near,far)` function:

```
          X axis: -5.0 to 5.0                    Y axis: -5.0 to 5.0                    Z axis: -5.0 to 5.0
left : -5.0                    5.0| bottom: -5.0                    5.0| near : -5.0                    5.0
right: -5.0                    5.0| top   : -5.0                    5.0| far  : -5.0                    5.0
```

Source: https://learnwebgl.brown37.net/08_projections/projections_ortho.html

Example 2/2



An orthographic projection demo.

Manipulate the parameters of the `createOrthographic(left,right,bottom,top,near,far)` function:

```
              X axis: 0.0 to 5.0                   Y axis: -5.0 to 5.0                     Z axis: -5.0 to 5.0
left : -5.0                        5.0 | bottom: -5.0                     5.0 | near : -5.0                     5.0
right: -5.0                        5.0 | top   : -5.0                     5.0 | far  : -5.0                     5.0
```

Source: https://learnwebgl.brown37.net/08_projections/projections_ortho.html

to make a camera effective

// to make a camera effective set the camera atribute of CGFscene

```
this.camera = c
```

// Recomputes the projection matrix, taking into account the canvas dimensions and the **active camera's parameters**. Can be called explicitly, or implicitly when e.g., the window is resized.

```
this.updateProjectionMatrix();
this.loadIdentity();
```

// Apply transformations corresponding to the **active camera position relative to the origin**

```
this.applyViewMatrix();
```

to make a camera change on mouse handlers

```
// assuming myInterface is an object of extending CGFinterface and myScene if an object extending CGFscene
myInterface.setActiveCamera(myScene.camera);
```