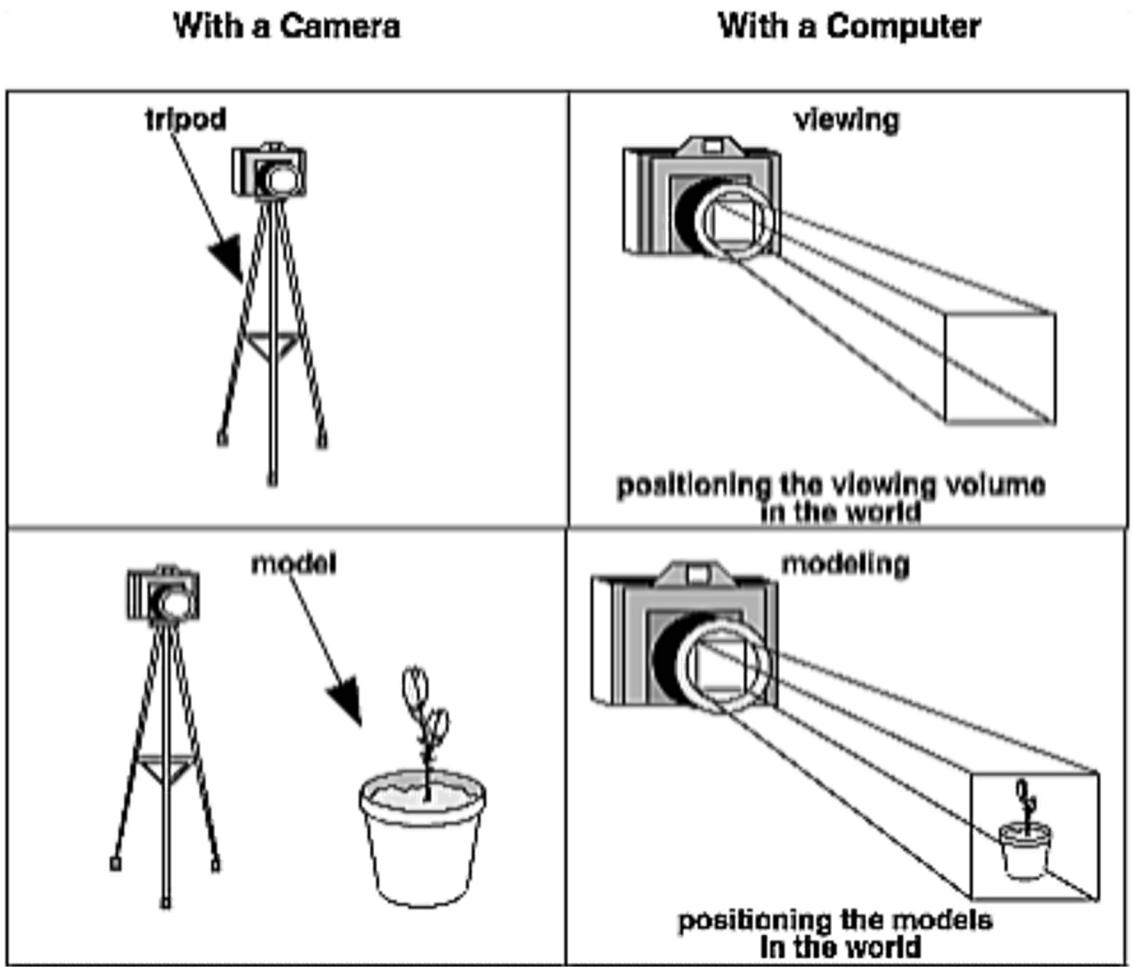
Lucerne University of
Applied Sciences and Arts
HOCHSCHULE
LUZERN
Informatik

WebGL: Viewing

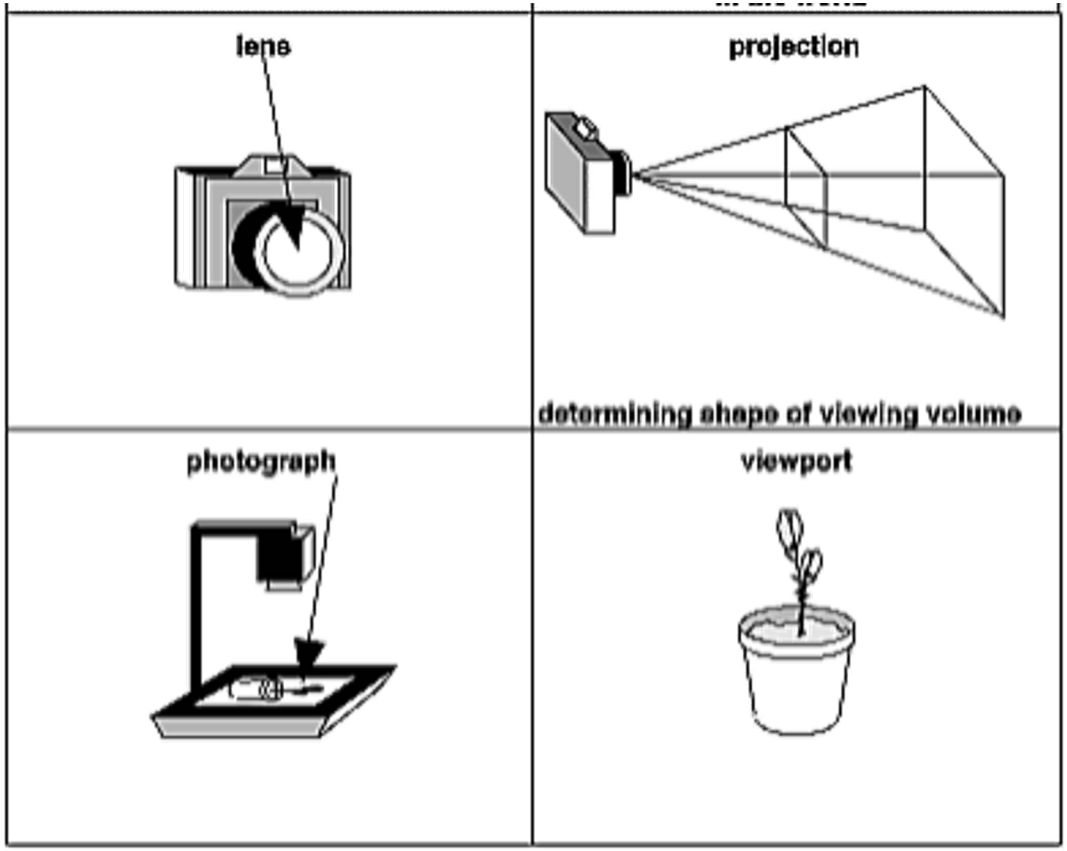
Informatik
Prof. Dr. Thomas Koller
Dozent

T direkt +41 41 349 35 38
thomas.koller@hslu.ch

Kamera Analogie

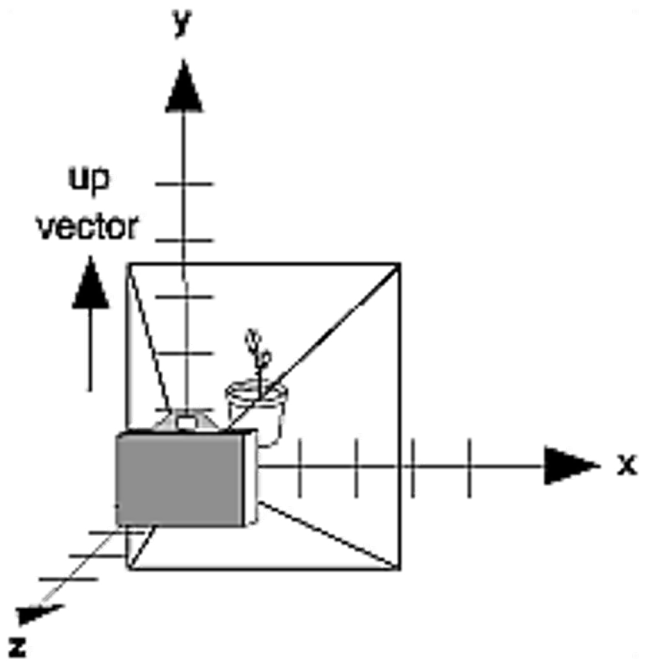


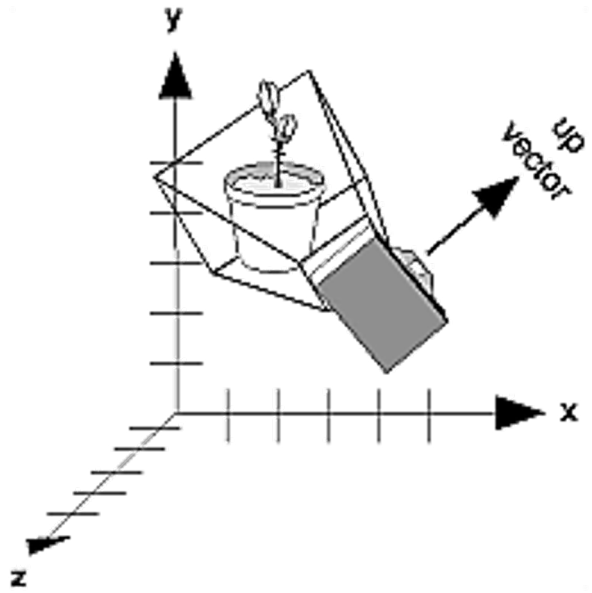
Kamera Analogie 2



Spezifikation der Kamera Position

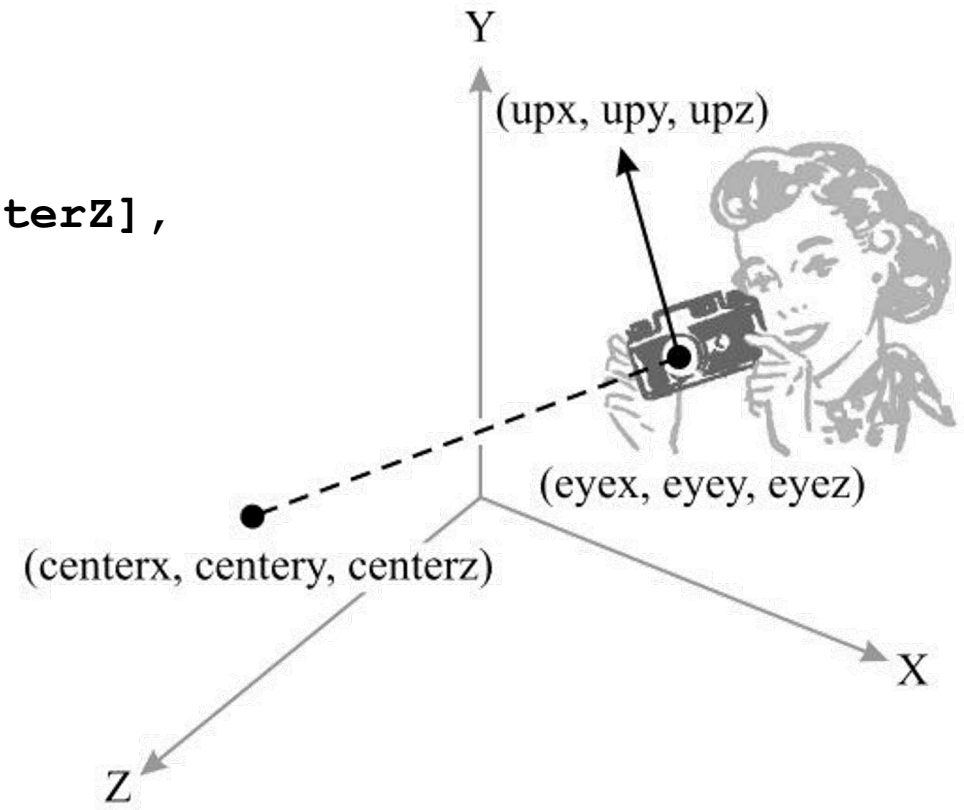
- Teil der ModelView Matrix
- Mittels „Transformationen“ oder mat4.lookAt(...)





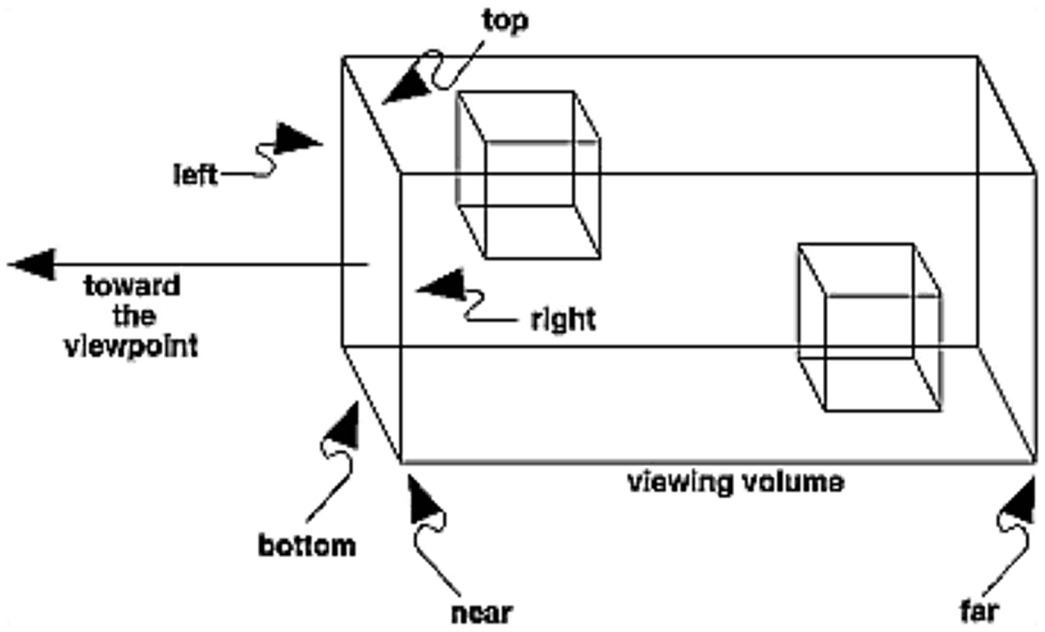
glMatrix lookAt()

```
var matrix = mat4.create();
mat4.lookAt(
   matrix,
   [eyeX, eyeY, eyeZ],
   [centerX, centerY, centerZ],
   [upX, upY, upZ]);
```



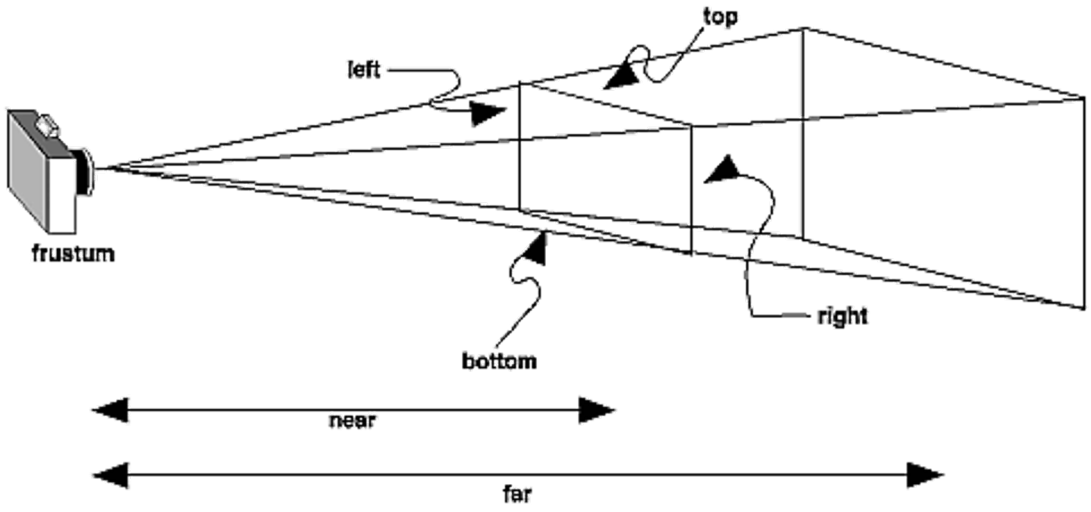
glMatrix Orthographische Projektion

```
var projectionMatrix = mat4.create();
mat4.ortho(projectionMatrix,
           left, right, bottom, top, near, far);
```



glMatrix Perspektivische Projektion I

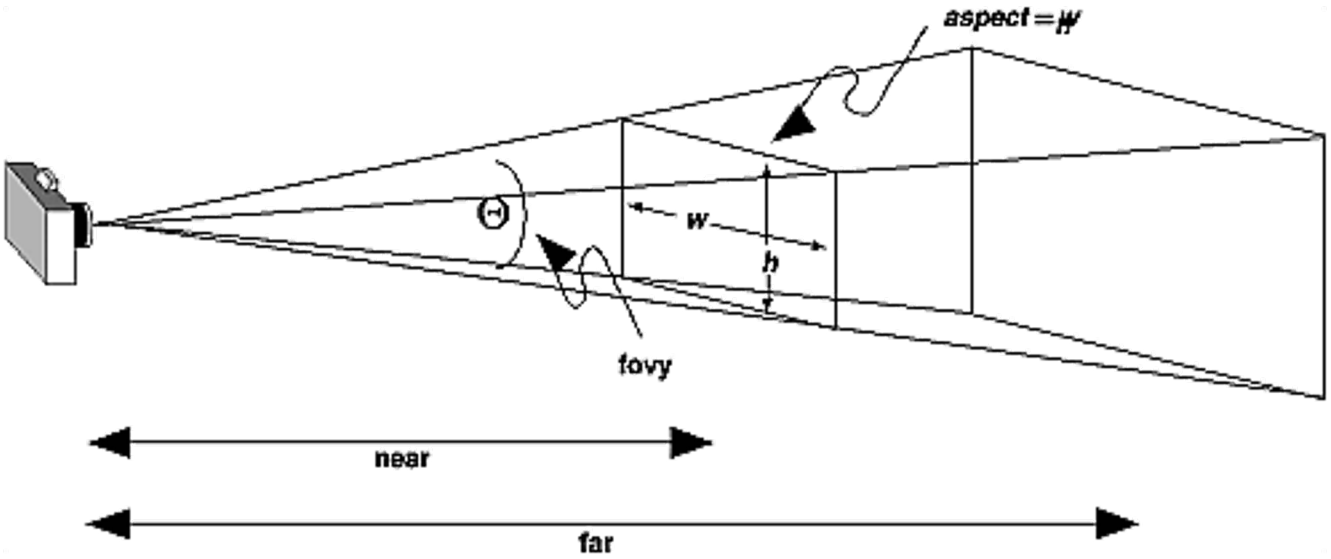
```
var projectionMatrix = mat4.create();
mat4.frustum(projectionMatrix,
             left, right, bottom, top, near, far);
```



near muss > 0 sein !

glMatrix Perspektivische Projektion 2

```
var projectionMatrix = mat4.create();
mat4.perspective(projectionMatrix,
                 fovy, aspect, near, far);
```



Spezifikation der Projektion

- Projektionsmatrix in Vertex Shader Program

```
attribute vec3 aVertexPosition;
uniform mat4 uModelViewMatrix;
uniform mat4 uProjectionMatrix;
void main() {
    vec4 position = vec4(aVertexPosition, 1.0);
    gl_Position =
       uProjectionMatrix * uModelViewMatrix * position;
}
```

Viewport

- Bestimmt wo im Fenster das Bild angezeigt wird

```
gl.viewport(x, y, width, height);
```

- Ganzes Fenster

```
gl.viewport(0, 0, windowWidth, windowHeight);
```