20 – picking and interaction

Input for Interaction

```
var clickStart: MousePosition | null = null;
var mousePosition: MousePosition | undefined = undefined;

canvas.onmousedown = (ev: MouseEvent) => { }
canvas.onmouseup = (ev: MouseEvent) => { }
canvas.onmousemove = (ev: MouseEvent) => { }
canvas.onmouseout = (ev: MouseEvent) => { }
```

Polling vs Asynchronous Events

How to Select

- Rays and Pixels: CPU vs GPU
- https://threejsfundamentals.org/threejs/lessons/threejs-picking.html

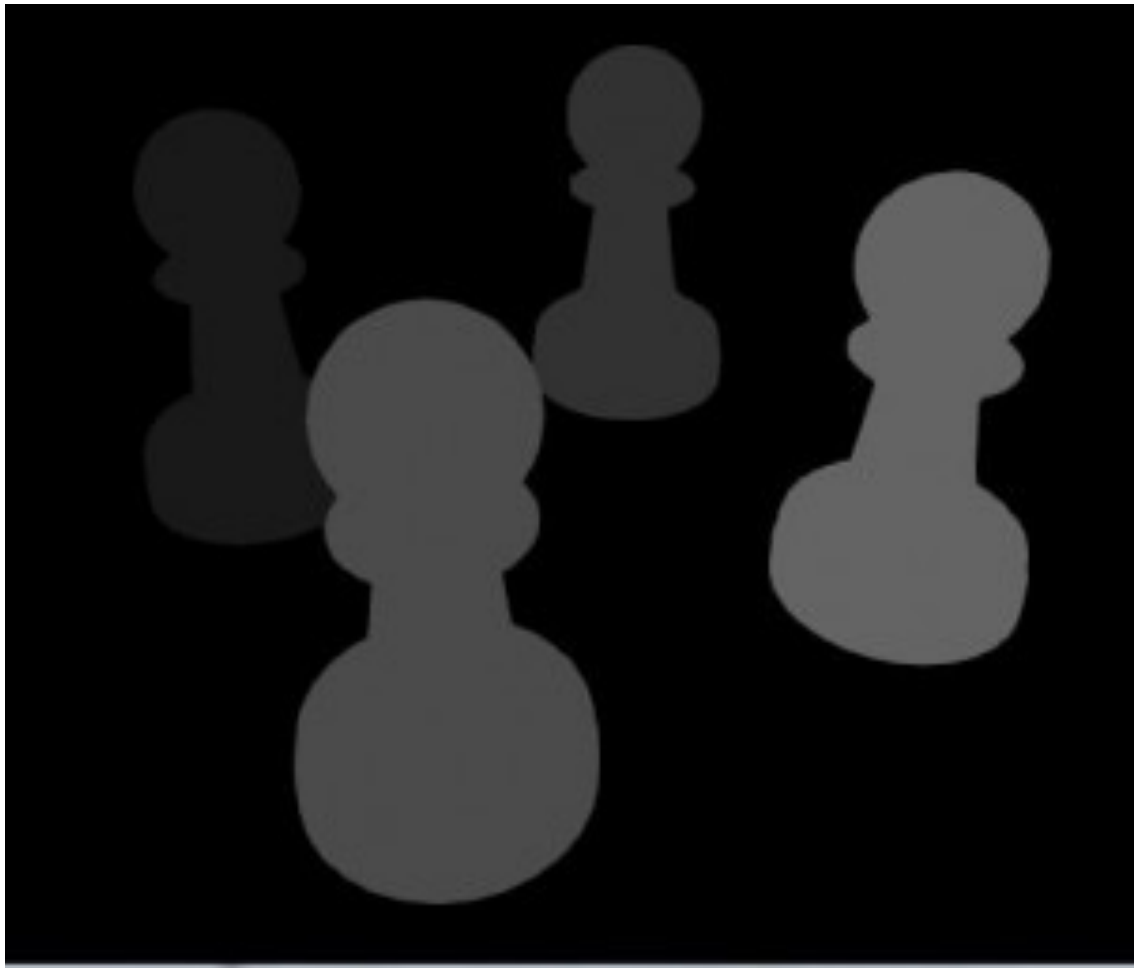
Pixel Based

http://voxelent.com/html/beginners-guide/chapter_8/ch8_Picking.html

http://learnwebgl.brown37.net/11_advanced_rendering/selecting_objects.html

https://www.sixhat.net/webgl-3d-picking-p5js-color-buffer.html

https://bl.ocks.org/duhaime/1eafa293e7ce16b074a6d55cac67badc



http://www.lighthouse3d.com/tutorials/opengl-selection-tutorial/

Pixel-based picking: three.js

```
pickingTexture = new THREE.WebGLRenderTarget(w, h);
canvas.addEventListener('mousemove', function(e) {
        renderer.render(pickingScene, camera, pickingTexture);
        var pixelBuffer = new Uint8Array(4);
        renderer.readRenderTargetPixels( pickingTexture, e.clientX,
                pickingTexture.height - e.clientY, 1, 1, pixelBuffer );
        var id = (pixelBuffer[0]<<16)|(pixelBuffer[1]<<8)|(pixelBuffer[2]);
}
// better:  make target 1,1 and use setViewOffset
```

Raycasting: three.js

```
raycaster = new THREE.Raycaster();
raycaster.setFromCamera(normalizedScreenPosition, camera);
intersectedObjects = raycaster.intersectObjects(scene.children);
```

Basic Code Structure

State Machines

e.g., https://github.com/eonarheim/TypeState