Revision of Simple Special Effects

Shanfeng Hu (Lecturer)

shanfeng2.hu@northumbria.ac.uk

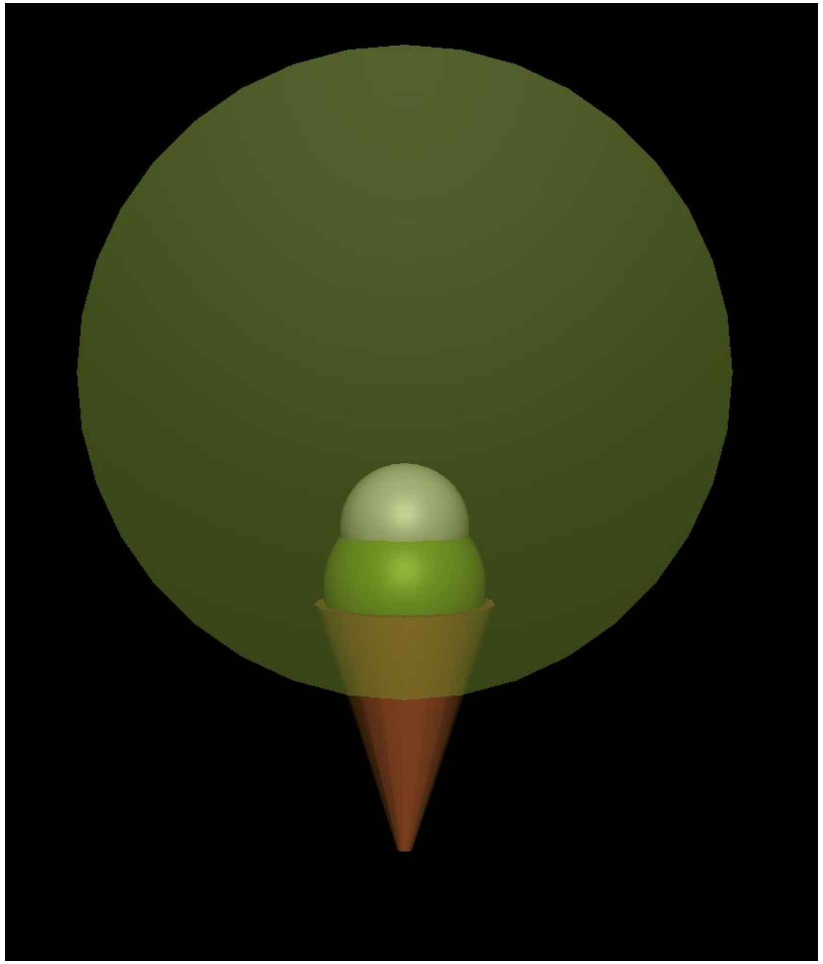
Department of Computer and Information Sciences
Northumbria University

Revision - Particle Systems

- A large number of small particles having different
  - Shapes
  - Colors
  - Transparencies
  - Positions
  - Movements
- Movement can be controlled by
  - Target locations
  - Movement directions
- Randomness for dynamic effects

Revision - Semi-Transparent

```
// A circle
var g4 = new THREE.CircleGeometry(1, 36);
var m4 = new THREE.MeshPhongMaterial({ color: 0x88AA22, transparent: true, opacity: 0.3});
var meshCircle = new THREE.Mesh(g4,m4);
meshCircle.position.z = 5.0;
scene.add(meshCircle);
```



Revision - Frame Number

- New a global variable to count the current number of frame
- Useful for animation

```
var iFrame = 0;
function animate()
{
   requestAnimationFrame(animate);
   iFrame ++;
   renderer.render(scene, camera);
}
```

Revision - Sin/Cos Function for Periodic Movements

```
var iFrame = 0;
function animate()
{
    requestAnimationFrame(animate);
    meshCircle.position.x = Math.sin(iFrame/100 + 3.14) * 10;
    iFrame ++;
    renderer.render(scene, camera);
}
```

The range of the periodic movement

The initial position of the periodic movement

The speed of the periodic movement

Revision - Array of Meshes - Creating

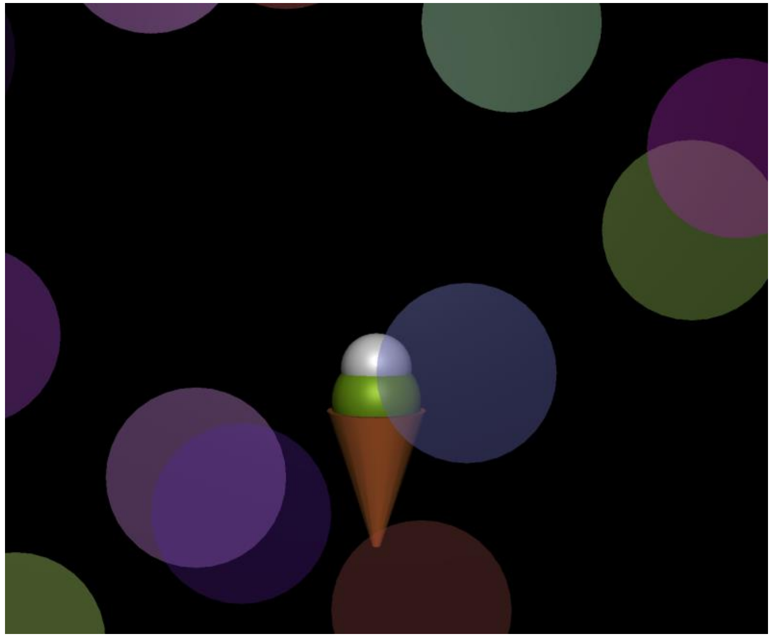
```
var geoCircleArray = [];
var matCircleArray = [];
var meshCircleArray = [];
var iCircleNumber = 5;
for (var i =0; i<iCircleNumber; i++)
{
   geoCircleArray.push( new THREE.CircleGeometry(1, 36) );
   matCircleArray.push( new THREE.MeshPhongMaterial({ color: 0x88AA22, transparent: true, opacity: 0.5 }) );
   meshCircleArray.push( new THREE.Mesh(geoCircleArray[i], matCircleArray[i]));
   meshCircleArray[i].position.z = 5.0;
   meshCircleArray[i].position.x = i;
   scene.add(meshCircleArray[i]);
}
```

Revision - Array of Meshes – Moving

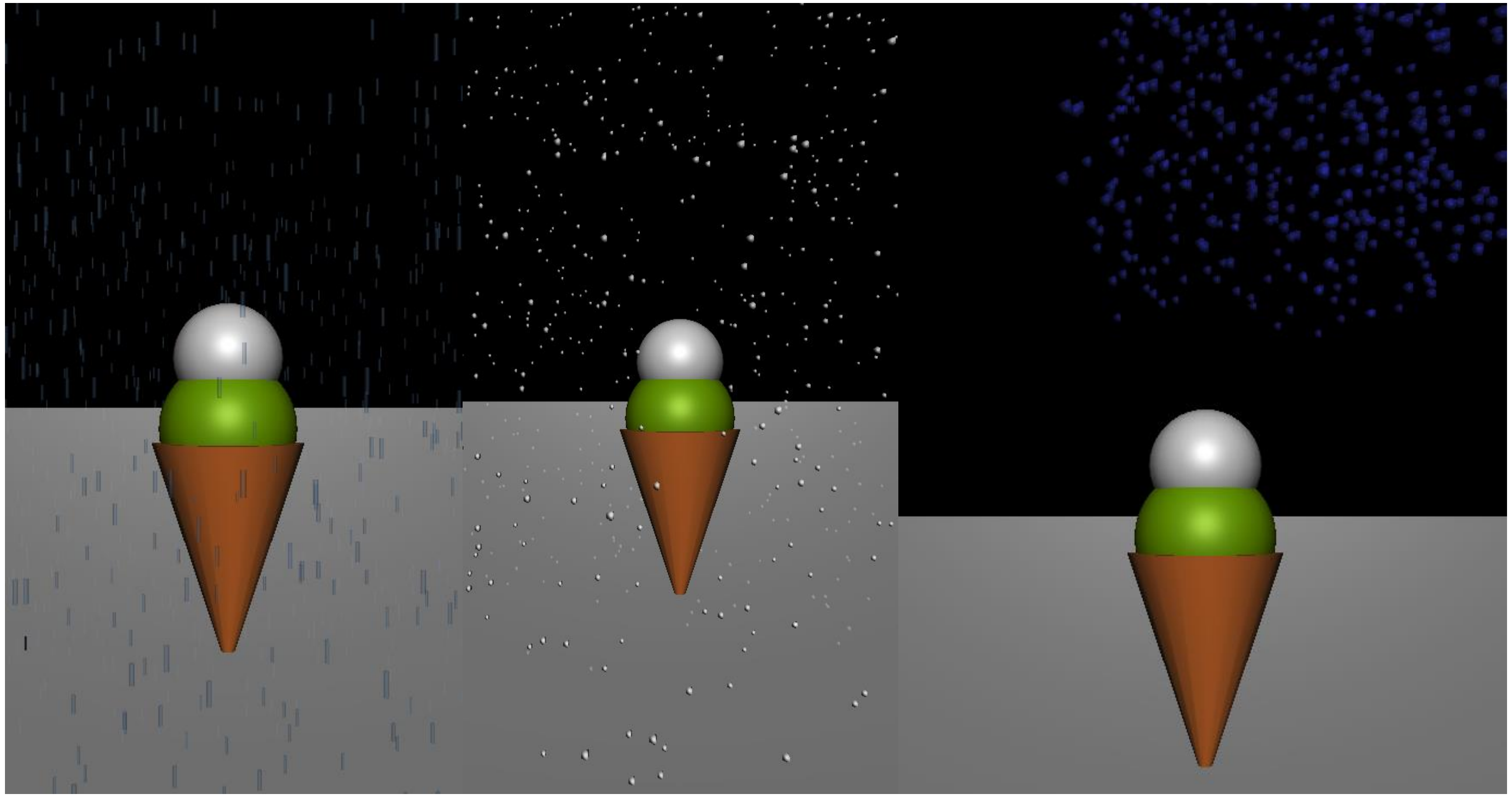
```
function animate()
{
    requestAnimationFrame(animate);
    for (var i =0; i<iCircleNumber; i++)
    {
        meshCircleArray[i].position.y = Math.sin(iFrame/200 + i*10) * 6;
    }
     iFrame ++;
    renderer.render(scene, camera);
}
```

Revision - Random Function

- Math.random()
  - Returns a value between 0 and 1
  - 10 * Math.random() returns a value between 0 and 10
  - 10 * Math.random() – 5 returns a value between -5 and 5
- e.g. Randomize the color of the circles
  - color: Math.random() * 0xFFFFFF
- e.g. Randomize the opacity
  - opacity: Math.random() * 0.2 + 0.3



Simple Particle Effects



Further Reading

- three.js Function List & Basic Tutorials
  - https://threejs.org/docs/#manual/en/introduction/Creating-a-scene
- A Very Good Tutorial on Particle Systems
  - https://aerotwist.com/tutorials/creating-particles-with-three-js/

The End

Any Questions?