

WebGL Translate, Rotate, Scale

R. B. Wishnumurti 13519203 K4

[Link Repository Github](#) | [Link Video Youtube](#)

Proses scale pada webgl adalah dengan mengkalikan posisi dengan nilai skala yang diinginkan. Pada tabel dibawah diperlihatkan bahwa posisi awal akan dikalikan dengan nilai skala untuk melakukan proses translasi dan rotasi. Nilai yang akan diinput adalah posisi, nilai sudut, nilai scaleX, dan scale Y.

```
<!-- vertex shader -->
<script id="vertex-shader-2d" type="x-shader/x-vertex">
attribute vec2 a_position;

uniform vec2 u_resolution;
uniform vec2 u_translation;
uniform vec2 u_rotation;
uniform vec2 u_scale;

void main() {
    // Scale the position
    vec2 scaledPosition = a_position * u_scale;

    // Rotate the position
    vec2 rotatedPosition = vec2(
        scaledPosition.x * u_rotation.y + scaledPosition.y * u_rotation.x,
        scaledPosition.y * u_rotation.y - scaledPosition.x * u_rotation.x);

    // Add in the translation.
    vec2 position = rotatedPosition + u_translation;
```

Lalu di javascript kita akan melakukan ke nilai-nilai tersebut dan membuatnya lebih mudah digunakan dengan adanya sliders.



```

// Setup a ui.
    webglLessonsUI.setupSlider("#x", {value: translation[0], slide: updatePosition(0), max:
gl.canvas.width });
    webglLessonsUI.setupSlider("#y", {value: translation[1], slide: updatePosition(1), max:
gl.canvas.height});
    webglLessonsUI.setupSlider("#angle", {slide: updateAngle, max: 360});
    webglLessonsUI.setupSlider("#scaleX", {value: scale[0], slide: updateScale(0), min: -5,
max: 5, step: 0.01, precision: 2});
    webglLessonsUI.setupSlider("#scaleY", {value: scale[1], slide: updateScale(1), min: -5,
max: 5, step: 0.01, precision: 2});

function updatePosition(index) {
    return function(event, ui) {
        translation[index] = ui.value;
        drawScene();
    };
}

function updateAngle(event, ui) {
    var angleInDegrees = 360 - ui.value;
    var angleInRadians = angleInDegrees * Math.PI / 180;
    rotation[0] = Math.sin(angleInRadians);
    rotation[1] = Math.cos(angleInRadians);
    drawScene();
}

function updateScale(index) {
    return function(event, ui) {
        scale[index] = ui.value;
        drawScene();
    };
}

.
.
.
// set the resolution
gl.uniform2f(resolutionLocation, gl.canvas.width, gl.canvas.height);

```

```
// set the color
gl.uniform4fv(colorLocation, color);

// Set the translation.
gl.uniform2fv(translationLocation, translation);

// Set the rotation.
gl.uniform2fv(rotationLocation, rotation);

// Set the scale.
gl.uniform2fv(scaleLocation, scale);

// Draw the geometry.
var primitiveType = gl.TRIANGLES;
var offset = 0;
var count = 18; // 6 triangles in the 'F', 3 points per triangle
gl.drawArrays(primitiveType, offset, count);
}
}
.
.
.
```

Javascript akan melakukan penggambaran mengikuti dengan nilai yang diinput pada sliders.