

3D 터레인 생성 및 물체의 상호작용에 대한 보고서

학생 (6조) 김 진 혁^{조장}, 남 태 승, 박 윤 상
교수 심 재 창
학과 안동대학교 컴퓨터공학과
e-mail kjh030529@gmail.com

요약

본 보고서는 p5.js에서 퍼린노이즈를 활용하여 일치하는 UV의 색상값에 따라 버텍스 높이를 조절하는 방식으로 3D터레인을 생성하고 마우스의 좌표에 따라 이동하는 구체를 생성함으로써 Vertex와 노이즈 및 전반적인 3D 그래픽스에 대한 이해를 위해 작성되었다.

1 서론

본 보고서에서는 p5.js에서 퍼린 노이즈를 사용하여 터레인을 생성하는 방법을 소개한다. 우선 퍼린 노이즈의 개념과 작동 방식에 대해 설명한 후, 이를 활용하여 다양하고 동적인 터레인을 만들어내는 방법을 소개한다.

본 보고서는 OpenAI의 GPT3.0 모델을 사용한 ChatGPT 서비스를 이용하여 LaTeX 문법과 Overleaf 에디터, p5.js의 문법에 대해 공부하여 작성하였다. 또한 퍼린 노이즈와 버텍스의 정보 조사는 ChatGPT를 통해 조사한 자료를 바탕으로 Bing 검색엔진을 이용해 검증한 후 작성하는 형태로 진행되었다.

2 퍼린 노이즈

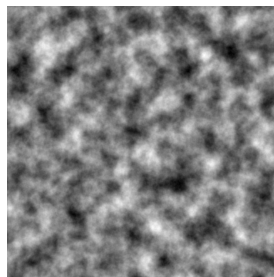


Figure 1: 퍼린 노이즈

퍼린 노이즈(Perlin noise)는 1983년에 Ken Perlin이 영화 '트론'을 제작중 컴퓨터 그래픽을 위한 단계적 질감 생성을 위해 개발한 그라데이션 노이즈의 한 종류로, 자연스러운 질감과 패턴을 만들어내기 위해 컴퓨터 그래픽과 절차적 콘텐츠 생성에서 널리 사용되고있다.

퍼린 노이즈의 기본 아이디어는 무작위로 방향이 정해진 벡터의 그리드를 생성하고, 이를 보간하여 부드럽고 연속적인 표면을 만드는 것이다. 이러한 노이즈 함수는 지형 등 다양한 자연스러운 패턴과 질감을 만들어내는 데 사용될 수 있다.

3 버텍스

버텍스는 그래픽스에서 가장 기본적인 개념 중 하나이다. 그래픽스에서의 모든 3D 모델은 버텍스들의 집합으로 이루어져 있다. 각 버텍스는 3차원 공간상의 한 점을 나타내며, 이 점은 3차원 좌표계 상에서 (x, y, z) 로 표현된다.

버텍스는 또한 모델의 다양한 속성을 포함할 수 있다. 예를 들어, 색상, UV, 법선 벡터 등의 정보를 가질 수 있다. 이러한 정보들은 모델을 입체적으로 렌더링하는 데 사용된다.

일반적으로 여러개의 버텍스 집합을 사용해서 삼각형 형태의 면을 표현한다. 이는 모델을 표현할 때 가장 일반적으로 사용되는 방법 중 하나이다.

4 구현

4.1 터레인 생성

p5.js에서 퍼린 노이즈를 사용하여 터레인을 생성하려면 2D 그리드의 퍼린 노이즈 값을 생성하고, 이를 이용하여 높이 맵을 생성한다. 아래는 이를 구현한 코드이다.

```
1  for (var y = 0; y < rows; y++) {
2    var xoff = 0, yoff = 0; //Terrain Scroll Speed
3    for (var x = 0; x < cols; x++) {
4      terrain[x][y] = map(noise(xoff, yoff), 0, 1, -50, 50);
5      xoff += 0.2;
6    }
7    yoff += 0.2;
8  }
```

그런 다음 삼각형의 메시를 생성하고 일정 각도를 기울이고 화면 중심으로 이동시킨다. 그리고 높이 맵을 사용하여 터레인의 각 지점의 높이를 지정하여 렌더링한다.

```
1  translate(0, 50);
2  rotateX(PI / 3);
3  translate(-w / 2, -h / 2);
4  for (let y = 0; y < rows - 1; y++) {
5    beginShape(TRIANGLE_STRIP);
6    for (let x = 0; x < cols; x++) {
7      let v = terrain[x][y];
8      vertex(x * scl, y * scl, terrain[x][y]);
9      vertex(x * scl, (y + 1) * scl, terrain[x][y + 1]);
10   }
11   endShape();
12 }
13
```

위 코드를 작성하여 실행하면 아래와 같은 화면을 볼 수 있다.

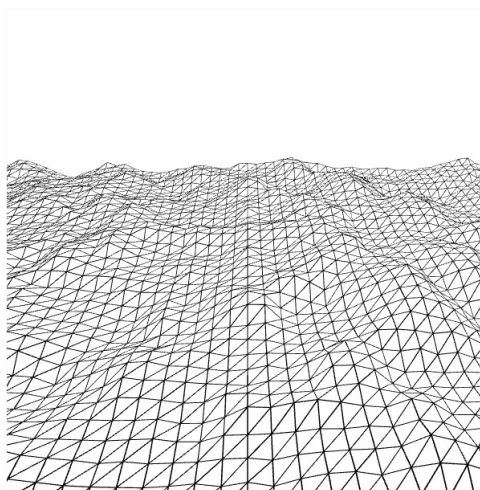


Figure 2: 터레인 생성

이제 터레인의 fill(int, int, int) 함수를 활용하여 높이에 따른 그라데이션 형식의 색을 면에 채운후 noStroke() 함수로 터레인의 선을 제거한다.

이후 마우스의 화면상의 좌표에 따라 터레인 위를 움직이는 구체에 대한 코드를 작성한다. 아래 코드에서는 구체의 입체적인 가시성을 위해 normalMaterial() 함수를 사용하여 3D 구체의 법선 값을 색상으로 표현하였다.

```

1  push(); translate(w / 2, h / 2);
2  translate(mouseX - width / 2, (mouseY - height / 2) * 6);
3  rotate(PI / 5);
4  normalMaterial(); pop();
5

```

위 코드를 사용하면 아래 보기와 같이 마우스 좌표에 따라 이동하는 구체가 생성된다.



Figure 3: 구체 생성

위의 코드를 모두 종합한 코드와 보기는 아래와 같다.

```

1  let cols, rows;
2  let scl = 20;
3  let w = 1400;
4  let h = 1000;
5
6  let flying = 0;
7
8  let terrain = [];
9
10 function setup() {
11
12   createCanvas(600, 600, WEBGL);
13   cols = w / scl;
14   rows = h / scl;
15   noStroke();
16
17   for (var x = 0; x < cols; x++) {
18     terrain[x] = [];
19     for (var y = 0; y < rows; y++) {
20       terrain[x][y] = 0; //specify a default value for now
21     }
22   }
23 }
24 function draw() {
25
26   flying -= 0.1;
27   let yoff = flying;
28   for (var y = 0; y < rows; y++) {
29     var xoff = 0;
30     for (var x = 0; x < cols; x++) {
31       terrain[x][y] = map(noise(xoff, yoff), 0, 1, -50, 50);
32       xoff += 0.2;
33     }
34     yoff += 0.2;
35   }
36
37   background(150,200,255);
38   translate(0, 50);
39   rotateX(PI / 3);
40   fill(200, 200, 200, 150);
41   translate(-w / 2, -h / 2);
42   for (let y = 0; y < rows - 1; y++) {
43     beginShape(TRIANGLE_STRIP);

```

```

45     for (let x = 0; x < cols; x++) {
46         let v = terrain[x][y];
47         fill(50 - (v+50)/100 * 50, (v+50)/100 * 255, 80 - (v+50)/100 * 80)
48         vertex(x * scl, y * scl, terrain[x][y]);
49         vertex(x * scl, (y + 1) * scl, terrain[x][y + 1]);
50     }
51     endShape();
52 }
53 push(); translate(w / 2, h / 2);
54 translate(mouseX - width / 2, (mouseY - height / 2) * 6);
55 rotate(PI / 5);
56 normalMaterial();
57 sphere(100); pop();
58 }
59

```

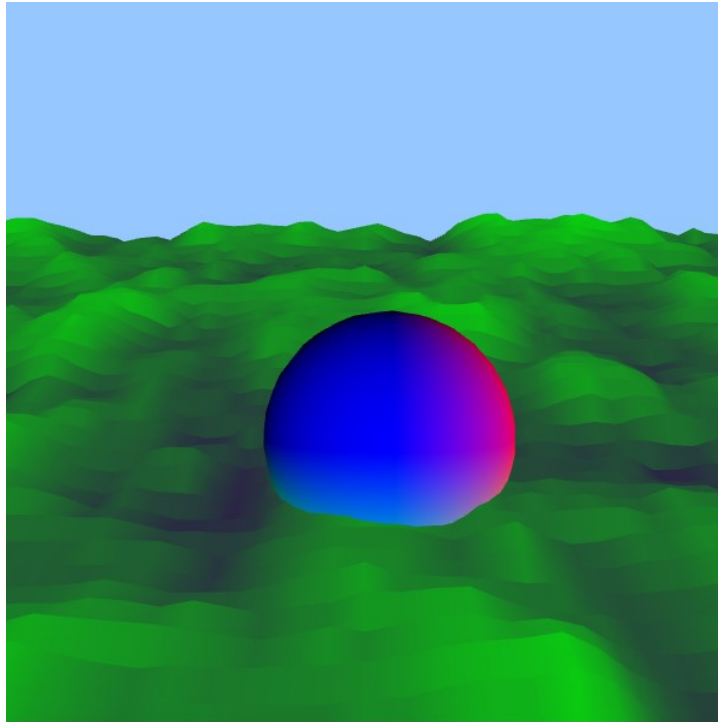


Figure 4: 결과

5 결론 및 기대

p5.js에서 퍼린 노이즈를 사용하여 터레이인을 생성하는 방법을 활용하여 움직이는 터레이인을 만들어내는 실습을 진행하였다. 이를 활용하여 다양한 그래픽 작업에서 유용하게 활용할 수 있을것으로 기대한다. p5.js에서는 이외에도 다양한 그래픽 기능을 제공하므로, 다양한 창의적 작업을 수행할 수 있을것으로 기대한다.