

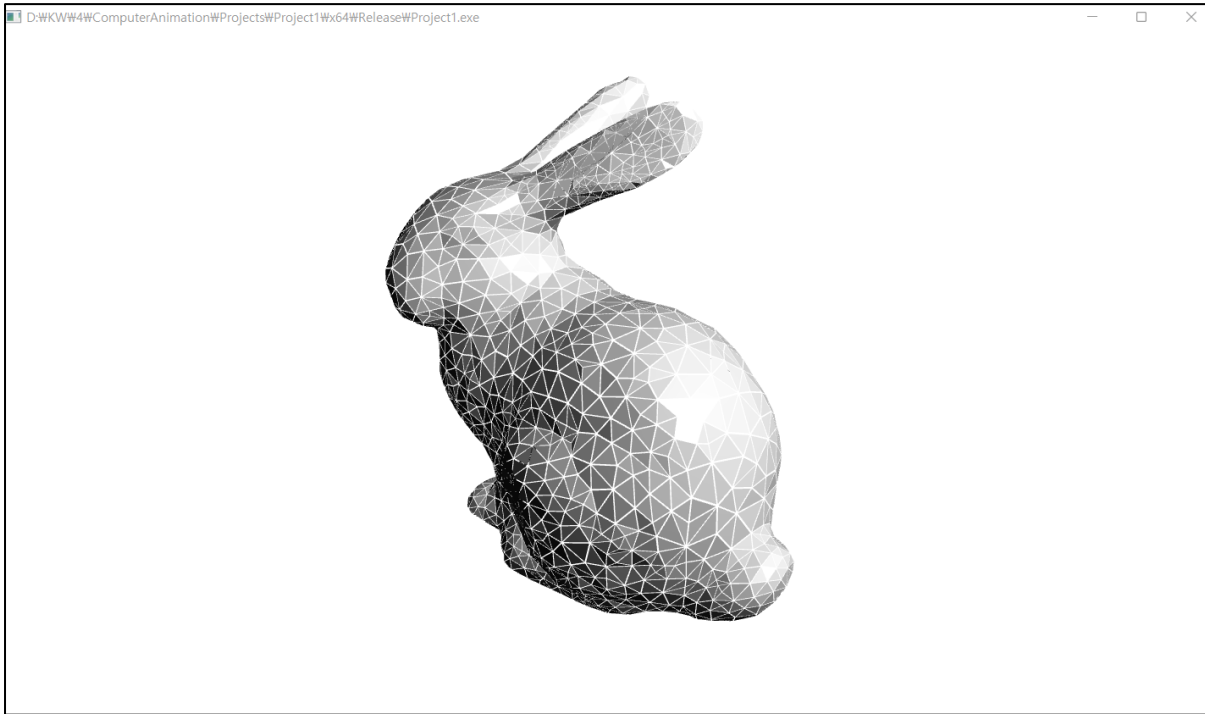
컴퓨터 애니메이션 실습 보고서



Self-Scoring Table

	P1	E1	Total
Score	1	1	2

P1 – Build and draw a mesh with shrunken faces



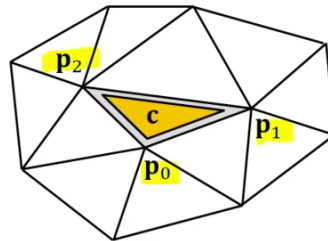
practice 1의 결과이다.

□ Center of mass of a triangle

$$\mathbf{c} = \frac{(\mathbf{p}_0 + \mathbf{p}_1 + \mathbf{p}_2)}{3}$$

□ Shrink vertices to COM

$$\blacksquare \Delta \mathbf{p}_i = \alpha(\mathbf{c} - \mathbf{p}_i)$$



Shrunken face를 만드려면 Vertex들의 중심점을 알아야한다. 중심점을 구하는 식은 위 그림을 통해 알 수 있다. 또한 α 값은 shrunken되는 정도다. α 값을 0.01을 주면 1퍼센트, 0.1을 주면 10퍼센트 하겠다는 의미다.

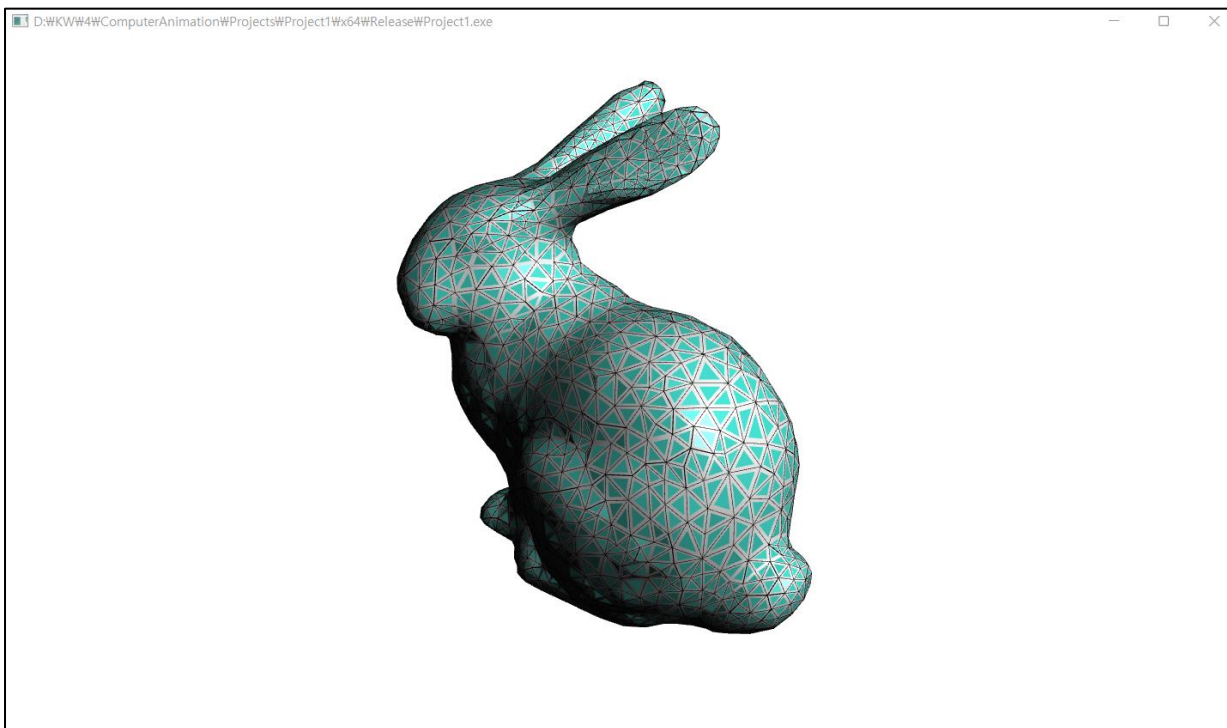
```

128 void buildShrunkenFaces(const MatrixXf& vertex, MatrixXf& faceVertex)
129 {
130     // Face mesh with # faces and (3 x # faces) vertices
131
132     // face마다 3개의 vertex가 필요.
133     // vertex가 공유되지 않고, triangle마다 독립되게 가져야하니깐용.
134     faceVertex.resize(3, 3 * face.cols());
135     for (int i = 0; i < face.cols(); i++)
136     {
137         //face.cols -> 컬럼 개수.
138
139         // Center of the i-th face
140         Vector3f center(0, 0, 0);
141         for (int j = 0; j < 3; j++)
142             center += vertex.col(face(j, i));
143
144         center /= 3.0f; // center of mass.
145
146         // Build 3 shrunken vertices per triangle
147         for (int j = 0; j < 3; j++)
148         {
149             const Vector3f& p = vertex.col(face(j, i));
150             faceVertex.col(3 * i + j) = p + gap * (center - p); // gap = 알파. 원래 피에서 센터방향으로 이동시켜주면 돼요.
151         }
152     }
153 }

```

코드로 구현한 부분은 위 그림에서 확인할 수 있다.

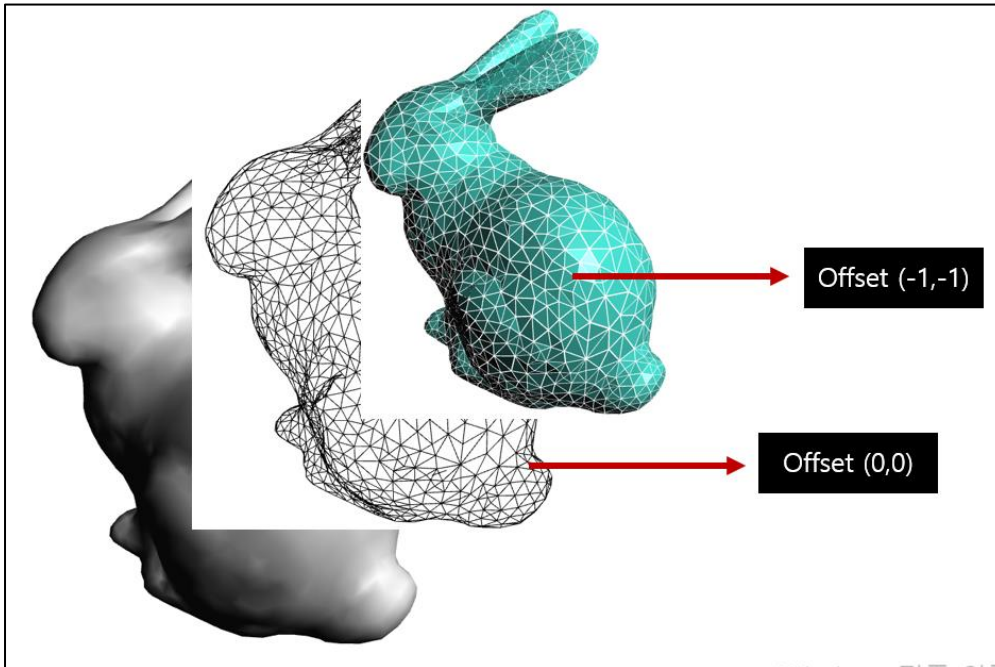
E1 – Overlay the shrunken faces on the original wireframed faces with offsetting.



Exercise 1의 결과이다.

- 구현 방법

1. 먼저 bunny mesh를 그린다.
2. 그 위에 offset을 주어서 wireframe을 그린다.
3. 그려진 bunny mesh와 bunny mesh의 wireframe 위에 offset을 주어서 shrunken face를 그린다.



- 코드

```
279 // Draw the mesh after setting up the material
280 if (faceWithGapMesh)
281 {
282     //원본 토끼를 그린다.
283     drawMesh();
284
285     // wireframe을 그린다.
286     glDisable(GL_LIGHTING); // Lighting을 끈다.
287     glEnable(GL_POLYGON_OFFSET_LINE); //Offset
288     glPolygonMode(GL_FRONT_AND_BACK, GL_LINE); // Polygon을 Line으로 그린다.
289     glPolygonOffset(0.0f, 0.0f); // offset을 0.0, 0.0으로 준다.
290     drawMesh(); //이 상태로 그리면 wireFrame되어있다.
291
292     glEnable(GL_LIGHTING); // 다시 Lighting을 켜다.
293     glDisable(GL_POLYGON_OFFSET_LINE);
294
295     // shurnkenFace를 그린다.
296     glEnable(GL_POLYGON_OFFSET_FILL);
297     glPolygonMode(GL_FRONT_AND_BACK, GL_FILL); // Polygon을 면을 채우게끔 그린다.
298     glPolygonOffset(-1.0f, -1.0f); // offset을 준다.
299     drawShurnkenFaces(); // 이 상태로 그리면 wireframe+원본 토끼 위에 그려진다.
300     glDisable(GL_POLYGON_OFFSET_FILL);
301
302 }
303 else drawMesh();
```

- 코드 설명

line 283 : drawMesh()를 이용하여 bunny mesh를 그린다.

line 285 ~293 : 이 코드들은 wireframe을 그리기 위한 코드다. glEnable(GL_POLYGON_OFFSET_LINE)으로 Polygon을 line 형태로 그릴 때에 offset 적용을 허용하게 한다. 그 다음 glPolygonMode(GL_FRONT_AND_BACK, GL_LINE)으로 폴리곤을 선 형태로만 그리게한다. glPolygonOffset(0,0)으로 offset을 설정한다. glPolygonOffset()은 음수 값이면 나에게 가까워지고 양수 값이면 나에게 멀어진다. 이 상태로 drawMesh()를 이용해 bunny mesh를 그리면 wireframe만 나오게 된다.

line 295 ~ 300 : 이 코드는 shrunken face를 그리기 위한 코드이다. glEnable(GL_POLYGON_OFFSET_FILL)으로 Polygon을 FILL 형태로 그릴 때에 offset 적용을 허용하게 한다. 그 다음 glPolygonMode(GL_FRONT_AND_BACK, GL_LINE)으로 폴리곤을 채우게끔 그리게한다. wireframe 보다 더 나에게 가까워야 하니, glPolygonOffset(-1,-1)으로 offset을 설정한다. 그리고 drawShrunkenFaces()를 통해 그린다.

- Reference

<https://www.codeproject.com/Articles/82/A-small-VRML-viewer-using-OpenGL-and-MFC>

<http://www.zeuscmd.com/tutorials/opengl/15-PolygonOffset.php>