

Graphics Hw3 Report

2013-11432 Cho Sung Min

0. 사용법

환경설정은 지난 과제들을 통해 이미 잘 되어있을 거라고 생각합니다. 실행 방법은 다음과 같습니다.

```
python3 main.py [input file 경로]
```

예를들어 실행 위치에서 input 파일이 input/umbrella3 이라면 실행은 다음과 같이 하면 됩니다

```
"python3 main.py input/umbrella3"
```

실행 되면 인풋 파일을 그린 모델이 보여지게 됩니다. 카메라 모듈을 붙여놓았으므로 돌아가면서 살펴보면 됩니다. 제가 제출한 모델은 umbrella3 로써 input 폴더 안에 있으므로

```
python3 main.py input/umbrella3
```

로 실행하면 됩니다.

1. Parsing

들어온 인풋 파일의 모든 라인에서 '#' 이 나타나는 지점 이후의 글자들은 지운 후에 라인들을 단어들로 split 해서 그 단어들의 stream 을 인풋으로 취급했습니다. 그 후 읽는 것은 straightforward 하게 했습니다.

2. CrossSection

CrossSection 클래스를 만들었습니다. 그것의 한 인스턴스를 생성할 때 컨트롤 포인트들, scale, rotation axis, rotation angle, translation vector 를 제공해주게 됩니다. 인스턴스를 생성하는 시점에 바로 그 정보들을 가지고 closed curve (b-spline 이거나 catmull_rom 일때 서로 다르게) 의 knot 들을 계산하고 그 knot 들이 scale, rotation, translation 을 통해 실제로 어디에 위치할지 계산해서 들고 있습니다. 계산하는 방법은 scale, rotation (이것은 quaternion 으로 만든 후 그것의 행렬을 곱해주는 방식), translation 을 GL 에서 한 후에 그것의 변환 행렬을 추출해서 (by glGetDoublev) 그 행렬을 knot 들에 곱해주는 식으로 계산했습니다. 후에 그림을 그릴 때는 이 knot 들의 실제 위치만 가지고 그리게 됩니다.

3. CrossSection 의 interpolation

CrossSection 의 interpolation 은 catmull_rom 으로 구현했습니다. scale 과 translation 은 straightforward 하게 interpolation 했습니다 (각 숫자를 catmull_rom 으로 interpolate). unit quaternion 의 경우가 까다로웠는데 rotation 과 axis 를 각각 따로 interpolation 한 후 합쳐서 만든 unit quaternion 을 interpolate 된 unit quaternion 으로 해석했습니다.

Catmull_Rom 으로 하기 때문에 양 끝의 cross section 은 그려지지 않는다는 단점이 있습니다. 따라서 양 끝에 기존의 끝 cross section 을 각각 3개씩 추가해줌으로써 끝의 cross section 도 그려지게 하였습니다

4. Mesh

두 CrossSection 을 주면 그 들의 knot 들의 실제 위치를 가지고 triangluar mesh 를 그려주는 함수를 구현했습니다. 이는 glBegin(GL_TRIANGLE_STRIP) 을 킨 후 두 CrossSection 의 knot 들을 번갈아 찍어주는 방법으로 구현했습니다. 전체 swept surface 를 그릴 때에는 CrossSetion 들을 interpolate 해서 얻은 전체 CrossSection 리스트에서 모든 인접해있는 두 CrossSection 에 대해 이 함수를 불러서 mesh 를 그려 주었습니다.

5. Rendering

glPolygonMode(GL_FRONT_AND_BACK, GL_LINE) 을 킴으로써 제가 그린 triangle mesh 들의 선들이 다 보이도록 하였습니다.

6. Camera

저번 과제에서 구현한 카메라 모듈을 붙임으로써 모델을 자유롭게 살펴볼 수 있도록 하였습니다. 컨트롤은 지난 번과 같습니다 (마우스로 왼쪽 드래그로 rotation, 오른쪽 드래그로 translataion, i/o 로 줌 인/아웃, k/l 로 돌리 인/아웃, r 로 see all).

7. Color

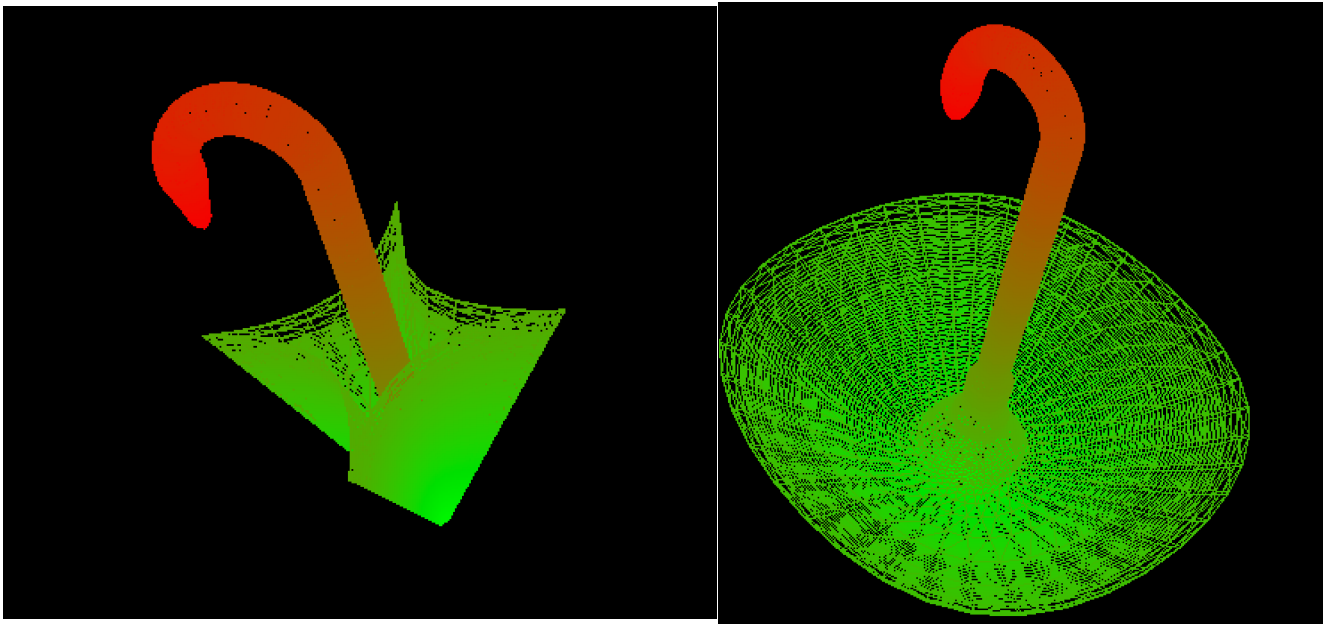
Swept surface 의 시작은 초록, 끝은 빨강이 되도록 그라데이션이 적용 되도록 하였습니다. 모델을 더 잘 이해하기 위해 시각적으로 도움이 되기 위함입니다.

8. Design

모델을 편리하게 만들기 위해 interactive 한 디자인 환경을 만들었습니다. cross section 을 추가하고 변형하고 움직이는 것을 interactive 하게 할 수 있게 해주는 환경으로써 이번 과제로써 제출하는 "뷰어" 위에 컨트롤 기능을 추가하여 만든 것입니다. 단순히 제 편의를 위해 만든 것이므로 이번 과제에는 "뷰어" 만 제출합니다.

9. 내 모델

우산을 만들었습니다. 3D 프린팅을 위해 우산의 꼭지와 손잡이의 끝 부분을 닫아줬습니다.



모델은 두가지가 있습니다. 날개가 동그란 것(input/umbrella3)과 날개가 뾰족한 것(input/umbrella2)이 있습니다. 3D printing 에 어떤 것이 더 유리할지 몰라서 두 개 다 보냅니다. 밑에 두 개의 결과 사진을 첨부합니다.