

HW3 보고서

컴퓨터공학부
2013-11413 우주

1. 실행 방법

Python 2.7 버전을 사용하였고 virtualenv로 격리하여 필요 패키지를 설치하였다. Python 2.7은 Ubuntu 16.04 에 기본 설치되어 있다. 실행은 아래 순서대로 진행하면 된다. 터미널에서 커맨드라인으로 실행해야 하는 부분은 앞에 \$ 를 써 놓았다.

- a. (pip가 설치되어 있지 않은 경우) \$ sudo apt-get install python-pip
- b. (virtualenv가 설치되어 있지 않은 경우) \$ pip install virtualenv
- c. 제출한 과제 디렉토리로 이동. requirements.txt 가 존재하는 폴더.
- d. (최초 실행시에만) \$ virtualenv .venv
- e. \$ source .venv/bin/activate
- f. (최초 실행시에만) \$ pip install -r requirements.txt
- g. \$ python HW3/HW3.py
 - 아무 인자를 주지 않고 실행하면 sample/my.txt 로 컵과 컵받침을 그린다.
 - 인자로 인풋 포맷에 맞는 파일을 넣어주면 해당 파일로 그린다.
 - ex) \$ python HW3/HW3.py HW3/sample/cokebottle.txt

2. 조작법

- a. 키보드 Up arrow : Zoom out.
- b. 키보드 Down arrow : Zoom in.
- c. 마우스 클릭 후 드래그 : Trackball rotation

3. 구현 내용 및 설명

a. Describe the control points of the cross section and transformations in the data file. The format is explained at the end of this assignment description. Your system should be able to parse the standard format.

- OK.

```
parsed = {  
    'curve_type': None, 'cross_sections': [],  
    'scalings': [], 'rotations': [], 'positions': [],  
}
```

curve_type 은 B Spline 인지 Catmull Rom 인 여부이다. cross_sections 는 입력으로 들어온 cross section 의 정보이다. points, scaling, rotation, position에 관한 정보가 들어있다. scalings 는 각 cross section 의 scaling 정보만 따로 모아둔 것이다. rotations 와 positions 도 마찬가지로 각각 rotation 정보 position 정보를 담고 있다.

b. Construct a closed curve using either B-splines or Catmull-Rom splines depending on what the data file demands.

- **OK.** cross section 의 첫 점과 마지막 점을 중첩하여 그려 closed curve 가 될 수 있게 하였다. 점과 점 사이 3개의 점을 추가하였다.

c. Describe a sequence of geometric transformations in the data file. Each transformation represents scaling, followed by rotation, followed by translation.

- **OK.** a 에서 설명한 대로 파싱하였고, 각 점을 X-Z 평면에 찍은 후 X, Z 축으로 scaling 후 quaternion rotation 을 진행하고 translation 해주었다.

d. Construct three splines for scaling factors, unit quaternions, and 3D positions.

- **OK.** 3가지 모두 Catmull Rom 으로 spline 을 만들었고 중간에 3개씩 추가해주었다. Quaternion 의 경우 SLERP 를 반복했고 SLERP 연산은 교수님의 mathclass 라이브러리를 파이썬으로 포팅했다.

e. Visualize the swept surface as a polygonal mesh. The rendering should be styled to present the shape of the surface clearly.

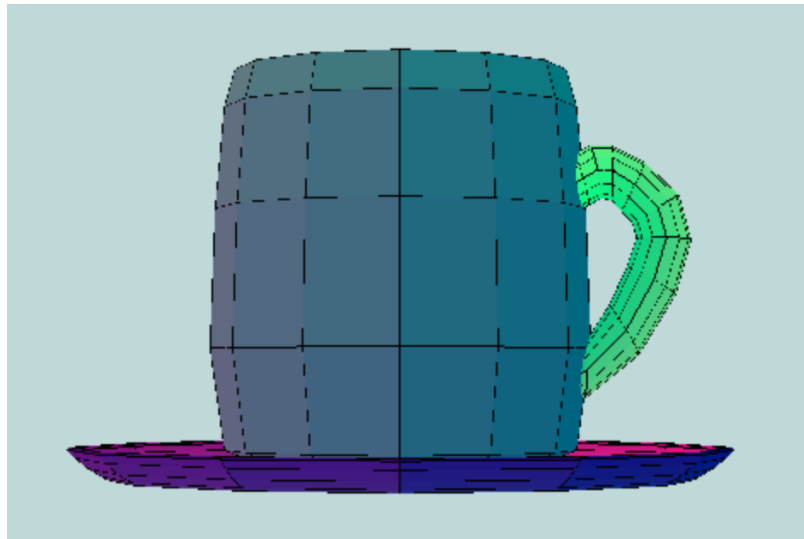
- **OK.** 아래 컵과 컵받침에서 볼 수 있듯이 사각형의 wire 를 표시해주었고 색을 조금씩 다르게 했다.

f. Allow for the user to rotate the scene so that we can inspect your surfaces at different viewpoints.

- **OK.** HW2 에서 구현한 Trackball rotation 을 사용하였다.

g. Create your own swept surfaces that are aesthetically pleasing.

- **OK.**



컵과 컵받침을 만들었다.