Computer Graphics Class Assignment 3

2015004493

김형순

1. How to run program

a. BVH 파일을 불러와 데이터를 화면에 rendering 하는 viewer 를 만들었습니다.

b. 프로그램은 총 4개의 .py 파일로 구성되어 있습니다.

b-1. camera.py : Class assignment 1 에서 구현한 camera 를 조작하는 코드 입니다.

b-2. lighting.py : Class assignment 에서 만들어놓은 코드를 사용한 shading 을 위한 조명을 사용 하기 위한 코드 입니다.

b-3. bvh\_viwer.py : bvh 파일을 parsing 하고, 화면에 rendering 하는 코드 입니다.

b-4. main.py : main 함수가 시작되는 코드 입니다. 위 세 코드를 import 해놓았습니다.

c. 프로그램 실행 방법 : 4개의 파이썬 파일 (camera.py, lighting.py, bvh\_viewer.py, main.py) 를 하나의 폴더에 넣은 후 main.py 를 실행 시킵니다.

d. 조작 방법

d-1. 실행 시킨 viewer 창에 열고 싶은 bvh 파일을 drag and drop 으로 열 수 있습니다.

d-2. 기본 설정은 motion data 없이 hierarchical structure (skeleton)을 line segment로 그렸습니다.

d-3. Space bar 를 누르면 line segment로 그려진 skeleton 이 motion data 에 따라 animated 됩니다.

**d-4.** Line segment 대신 box를 이용하여 skeleton을 rendering 할 수 있습니다. **A key** 를 누르면 line segment로 그려진 skeleton을 box를 이용한 skeleton 으로 바꿀 수 있습니다. 또한 box를 이용한 skeleton 은 lighting이 적용되어 shading이 된 것을 볼 수 있습니다.

2. Implemented requirements

a. Manipulate the camera : camera.py 에 구현되어 있습니다. Class assignment 1 code 를 사용하였습니다. Reference grid plane 은 main.py의 drawFrame 함수에 구현되어 있습니다.

b. Load a bvh file and render it

b-1. drag-and-drop

bvh\_viewer.py 에 drop\_callback 함수로 구현하였습니다. bvh 파일을 window 에 drop 하면 프로그램이 파일을 불러 올 수 있습니다.

b-2. Read the bvh file

bvh file 의 내부 정보를 parsing 하였습니다. drop\_callback, set\_hierarchy, set\_motion 함수에서 각 data들을 parsing 하여 hierarchy data 와 motion data 를 나눠 전역변수 hierarchy, motion 에 리스트 형태로 저장합니다. hierarchy data 는 각각 joint 가 [tag(R,J,E), offset, channel\_name, [child1], [child2] …] 형식으로 저장되고, motion data 는 [[First frame channels], [Second frame channels], …] 형태로 저장됩니다.

b-3. Print out the information

bvh file 을 성공적으로 불러오고 pasing 하였다면 File name, Number of frames, FPS, Number of joints, List of all joint names 가 stdout 으로 출력됩니다.

b-4. Render the “skeleton” (hierarchical structure)

Joint 의 시작점(부모와 연결된 부분)부터 offset 만큼 떨어진 점을 line segment 로 연결 합니다. End-effector 도 end-site 까지 line segment 로 연결 합니다. render\_hierarchy, draw\_hierarchy 함수에서 확인 할 수 있습니다. Line segment 로 skeleton 을 rendering 하는것은 lighting 을 적용하면 잘 보이지 않아서 lighting 과 shading 을 꺼두었습니다.

b-5. Animate the loaded motion

<spacebar> key 를 누르게 되면 key\_callback 함수에서 animate\_check 변수를 바꾸게 되고 default 인 false 에서 true 로 바뀌면 그때부터 skeleton 이 읽어온 motion data 에 따라 animate 됩니다. Spacebar 를 눌러 animate 를 시작하거나 animate 중 새로운 파일을 불러오면 항상 motion 의 가장 처음부터 실행 합니다.

b-6. About size of skeleton

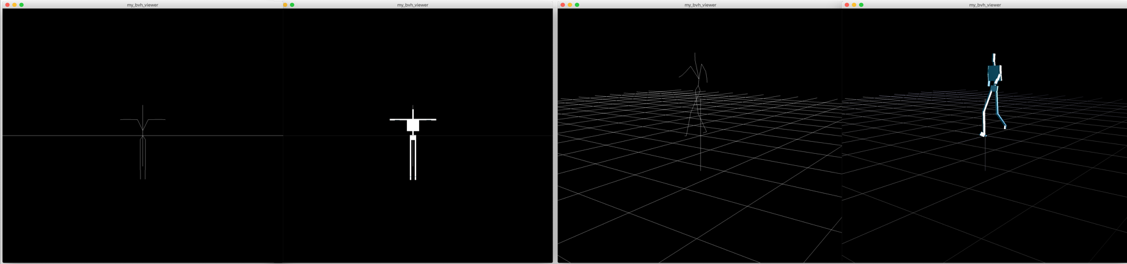
Offset 이 너무 커서 skeleton 이 화면에 다 들어오지 않았기 때문에 skeleton 의 size 를 최대 크기에 따라 조절 하였습니다. 원래 size 의 skeleton 을 원하면 S key 를 눌러 바꿀 수 있습니다. 그러나 size 가 너무 크면 animate 될 때 화면에 들어오지 않을 가능성이 큽니다. (sample-left-turn-larger.bvh 포함)

C. Extra credits A

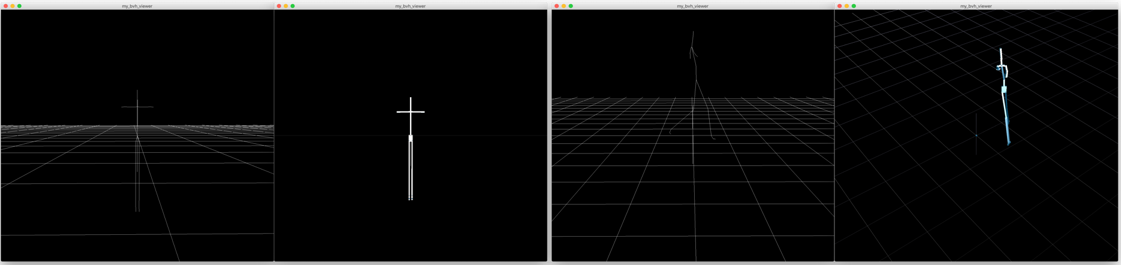
c-1 . Line segment 대신 box를 이용하여 skeleton 을 만들 수 있습니다. **A key** 를 눌러 line segment 와 box rendering mode 를 변경 할 수 있습니다.

c-2. Box 를 이용하여 skeleton 을 만들 때에는 lighting 과 shading 도 적용 시켜 줍니다. 조명의 위치는 Class assignment 2 의 조명 위치 그대로 사용 하였습니다.

3. Screen shots

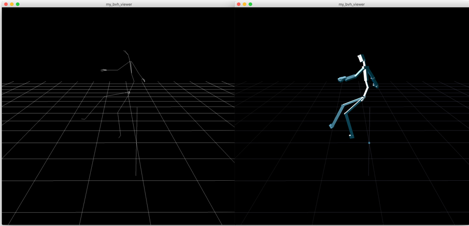
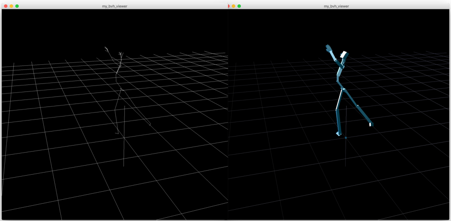
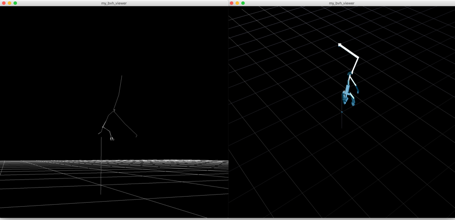
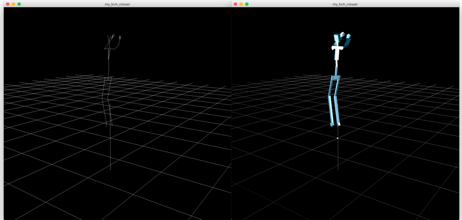


Sample walk



sample-left-

turn-larger

 Soccer Basket Ball 

Jump

dance