**컴퓨터 그래픽스 ClassAssignment3**

2019019016 서시언

**<implemented requirement>**

**A.** ClassAssignment1의 코드를 사용해, 같은 방식으로 카메라를 조작한다.

**B.** Load a bvh file and render it

1. glfw.set\_drop\_callback을 이용해 bvh viewer에 drag-and-drop을 통해 bvh file을 open한다.

- rendering하고 있던 bvh file이 있더라도 새로운 파일을 drag-and-drop하면 새로운 파일만을 rendering한다.

- readline(), split() 등을 이용해 bvh file을 parsing해서 Joint class type 변수에 각 root, joint, end site의 정보들을 저장한다.

- Joint class는 각 joint들의 offset, channel, name, child, parent, motion data 정보를 저장한다. (obj를 이용하는 경우 body part에 해당하는 obj file을 파싱한 결과 나온 varr 정보도 저장)

- root부터 child들을 tree 구조로 저장한다.

2. bvh file의 HIEARARCHY section을 이용해, joint간의 offset을 이용해 t-pose를 rendering한다.

- parent-child joint를 line segment로 연결한다.

- end-effector joint는 end site와 line segment로 연결한다.

- offset을 이용해 각 joint 및 end site의 위치를 계산한다. (translate, rotation은 identity로 설정)

3. <spacebar>를 누르면 bvh file의 MOTION을 load해서, 매 프레임마다 해당 pose를 rendering한다.

- glfw.swap\_interval(1)을 main function의 while문에서 호출한다.

- 마지막 프레임을 그리면 다시 첫 프레임부터 그린다.

- animating mode일 때 새로운 bvh file을 drag-and-drop하면 animating mode를 종료하고, 새로운 파일의 t-pose를 redering한다.

- animating mode에서 <spacebar>를 다시 누르면 아무일도 일어나지 않는다. (계속 animating mode)

- motion section에서 읽어온 데이터를 이용해 joint를 parent에 대해 transformation해서 위치를 계산한다.

- offset을 이용해 child와 연결한다.

4. bvh file을 open하면 file name, frame 수, FPS, root를 포함한 joint 수, 모든 joint의 이름 목록을 프린트한다.

**C.** Extra credit (B)

- drag-and-drop한 file이 ‘sample-walk.bvh’ 나 ‘sample-spin.bvh’인 경우, 각 body part에 대한 obj file을 load해서 line segment 대신 obj file을 reder한다.

- 이때, obj file에서 읽어온 normal을 이용하고 glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL )로 설정하고, glEnable(GL\_LIGHTING)을 해줘서 shading, lighting을 설정한다.

- 3개의 right source를 사용

- 빨간색, position: (10,10,10)

- 초록색, position: (-10,10,-10)

- 파란색, position: (10,10,-10)

**<Hyperlink to the video>**

- line segment

<https://youtu.be/ZYGRaP6dPhk>

- obj file (sample-spin.bvh)

<https://youtu.be/VkdGPVn4Kto>