

구현 사항

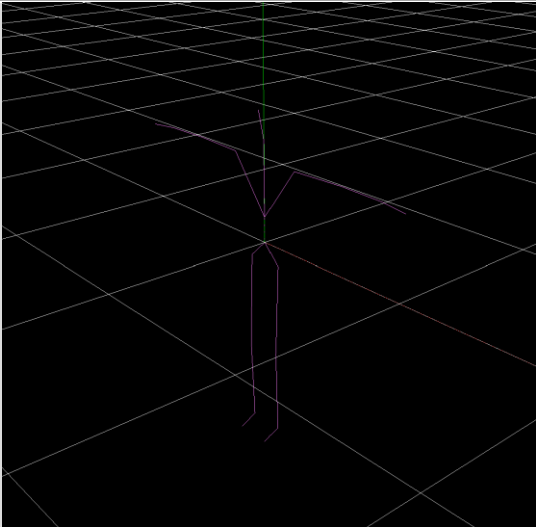
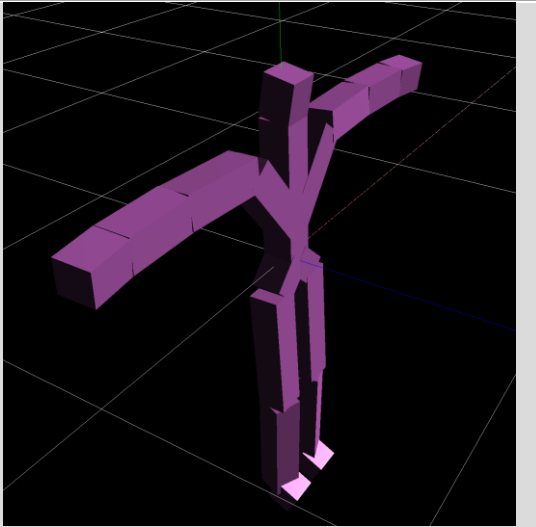
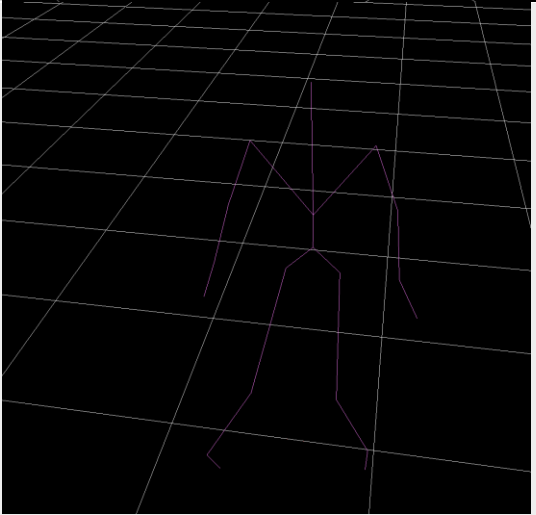
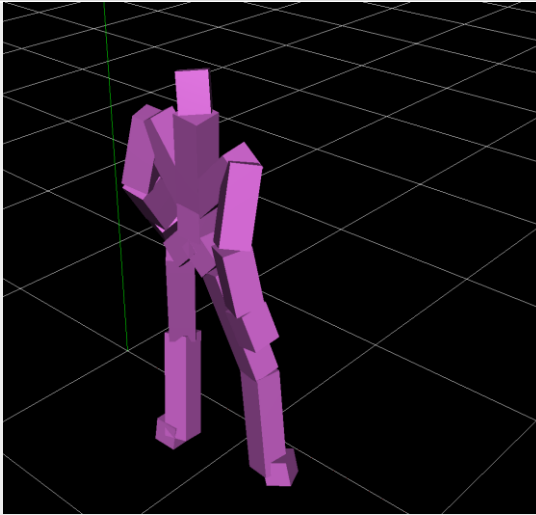
- A. Project1에서 구현했던 카메라 코드를 사용하여 카메라를 구현했고, reference grid plane(xz평면)을 그렸습니다. 추가적으로 x, y, z 축을 알아보기 쉽게 축을 그렸고, 카메라의 경우 기본적인 코드는 project1과 같으나 스크롤 속도나 카메라 rotating, panning할 때 좀더 빨리 움직이도록 sensitivity 상수값을 조금 올렸습니다.
- B. Load a bvh file and render it
 - i. bvh viewer window에 bvh file을 drag & drop 하여 bvh file을 열 수 있습니다. 오직 하나의 bvh 파일만 렌더링할 수 있으며, A.bvh 파일을 렌더링하다가 B.bvh 파일을 drag & drop하는 경우 B만을 새롭게 렌더링합니다.
 - [주요 파일 및 함수] main.py:drop_callback, loader.py 등
 - ii. '1'을 누르면 Line rendering mode(window를 처음 열었을 때의 옵션), '2'를 누르면 Box rendering mode로 진입할 수 있습니다. 모든 박스는 Phong Illumination과 Phong shading을 이용하여 렌더링됩니다. 프로그램 실행 중에는 언제든지 rendering mode를 바꿀 수 있습니다.
 - [주요 파일 및 함수] main.py:key_callback, g_fragment_shader_src, node.py:prepare_vao_line, prepare_vao_box 등
 - iii. bvh 파일을 drag & drop하여 로드한 직후에는, skeleton(rest pose)을 렌더링합니다. 애니메이션은 자동 재생되지 않으며, bvh 파일의 HIERARCHY section 데이터의 joint와 offset에 기반하여(즉, zero translation & rotation 상태) skeleton을 그립니다. end-effector joint역시 line으로 연결됩니다.
 - [주요 파일 및 함수] loader.py:draw_animation 등
 - iv. spacebar를 누르면 로드된 모션이 animate됩니다. MOTION의 각 line의 frame data를 이용하여 motion의 pose를 그립니다. bvh 파일에 지정된 프레임 시간이 경과할 때마다 한 번에 한 프레임씩 진행합니다. 모든 프레임을 그리고 나면, 다시 처음 프레임부터 그리게 됩니다.
 - [주요 파일 및 함수] main.py:key_callback, loader.py:draw_animation 등
 - v. bvh file을 열면(window에 drag drop하면), bvh 파일의 file name / number of frames / FPS / number of joints / list of all joint names를 콘솔창에 출력합니다.
 - [주요 파일 및 함수] loader.py:print_bvh_data 등

How to run

submission.zip 파일을 압축 해제한 뒤, camera.py, loader.py, main.py, node.py를 한 디렉토리 아래에 둔 채 python main.py를 터미널에 입력하여 실행할 수 있습니다.

Youtube Link: <https://youtu.be/BSnJhBOejh8>

실행 사진

| | Line rendering (Pressing '1') | Box rendering (Pressing '2') |
|----------------------------------|--|---|
| First drop: Rest pose |  |  |
| Animating (Pressing spacebar) |  |  |