컴퓨터 그래픽스

이윤상 교수님

**Report for Class assignment3**

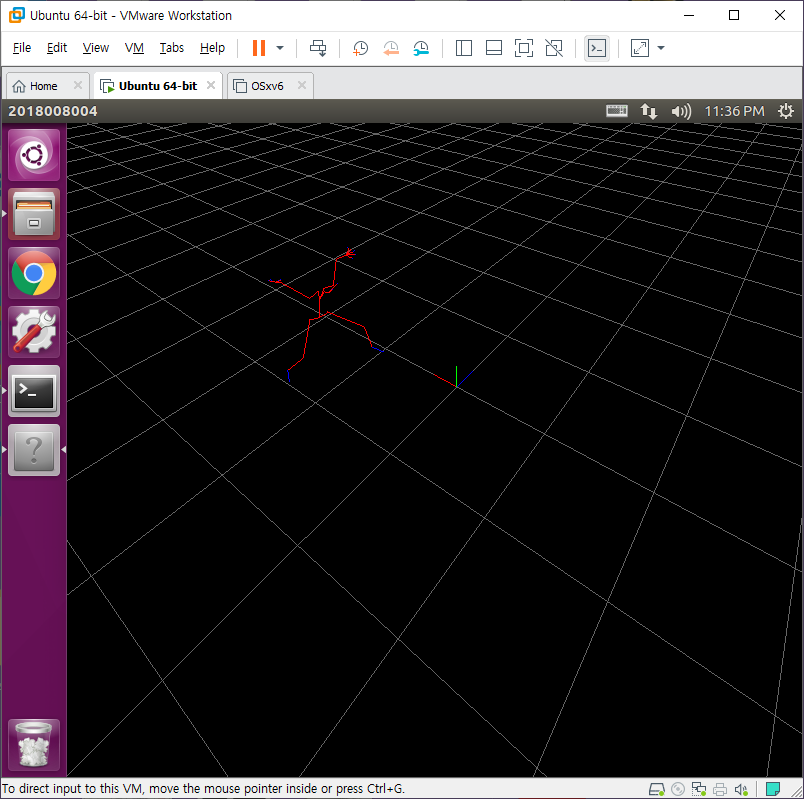
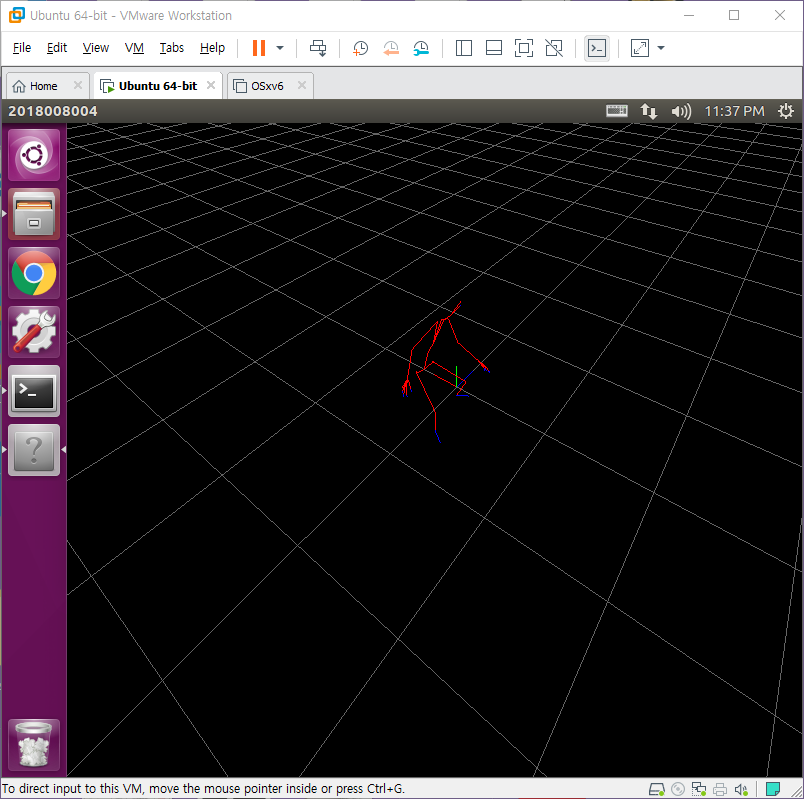
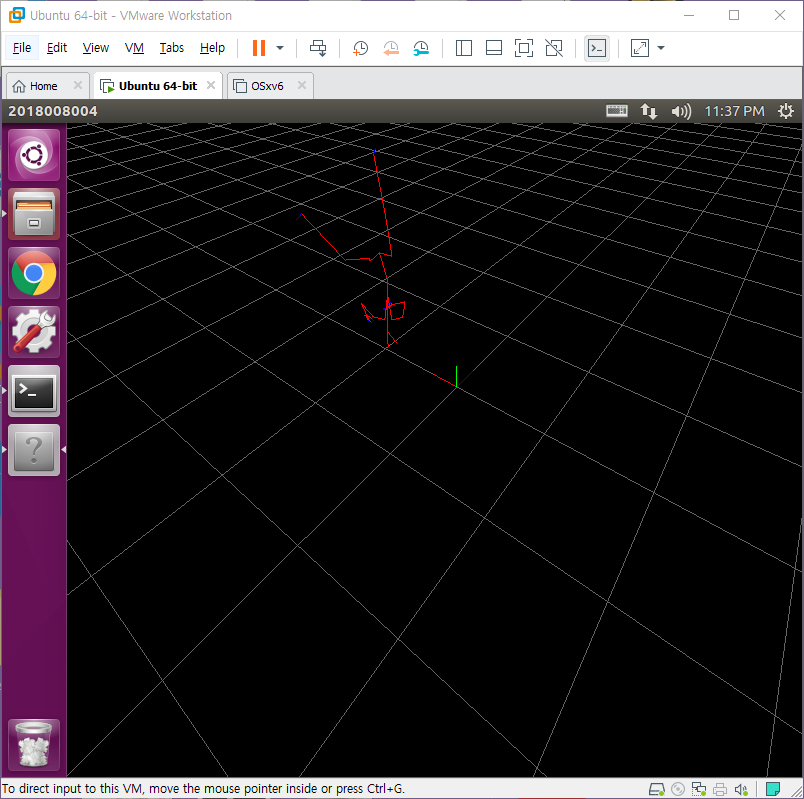
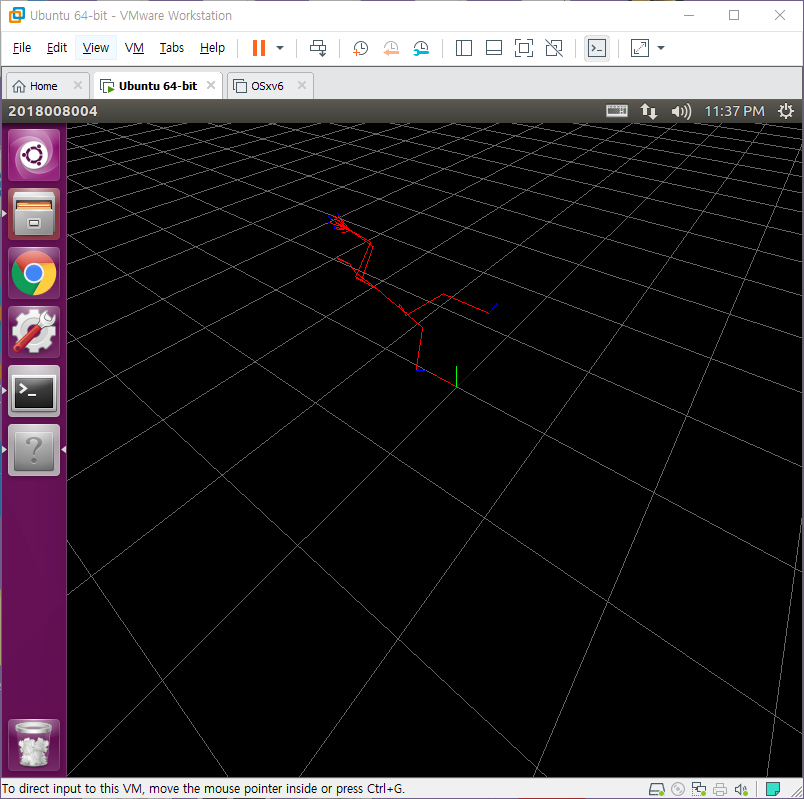
1. **Which requirements you implemented.**
2. Camera manipulation

과제1번의 코드를 skeleton code로 사용하였습니다.

1. Load an bvh file and render it
2. Drag-drop을 지원하기 위해 glfwSetDropCallback을 통해 file의 path를 얻어와 여는 방식을 사용했습니다. 다른 bvh file을 drag-drop시 이전에 bvhfile을 지우고 새로운 bvh file을 render합니다.
3. Bvh file을 읽어오면, HIERARCHY 부분과 MOTION 부분을 각각 parsing합니다. HIERARCHY 부분에서는 JOINT, CHANNELS, OFFSET을 global dictionary에 JOINT의 이름으로 CHANNEL과 OFFSET을 mapping하여 사용에 용이하게 했습니다. CHANNEL list와 MOTION의 column을 match하기 위하여 JOINT 이름과 column의 시작 index를 dictionary에 저장하였습니다. 마지막으로 parsing이 끝나면, tree형태의 BVH hierarchy를 gtree에 저장하여 사용합니다.
4. Parsing이 끝나면 gtree를 통하여 skeleton을 그립니다. 이 때, 모든 channel의 값이 0인 motion을 gMotion앞에 넣어 스페이스바 입력 이전까지 이 모습을 render합니다. 각 joint별 line segment를 보기 쉽게 색을 다르게 사용하였습니다.
5. 스페이스바를 입력하면 매 render마다 tick을 올려 motion을 rendering합니다. Motion의 길이로 tick을 modular하여 구현하였습니다.
6. Skeleton을 그릴 때는 gTree를 DFS로 탐색하며 각 joint를 그립니다.
7. Obj file을 drag-drop하여 읽어온 경우에 관련된 정보를 console에 출력했습니다.
8. Extra credits.

Extra credit 부분은 구현하지 않았습니다.

1. Screenshot of obj file render

7

<http://motion.hahasoha.net/> 085\_2 JumpTwist

*End of report*