

3D Computer Graphic Final Project

Xiaoxu Gao

Ruoqing Sun

Anelia Dimitrova

How to play:

Hero keeps walking on a 3D terrain height-field terrain with a horse which uses mesh simplification. He should control the direction of the bullet to shoot golden enemies. After all enemies die, the big boss with two guards will show up. After keeping shoot for a while, his partial vertices will disappear. In the end, the big boss will die and disappear.

Technique summary:

Hero:

1. Animation (video)
2. Texture mapping (Fig7) : reading "vt" from obj

Horse:

1. Mesh simplification (half-edge data structure with random edge-collapse)
Fig11

Bullet:

1. Transformation Matrix (video)
2. Texture mapping (Fig7)
3. Texture changing(video): change every frame

Enemies:

1. Phong shading (Fig7): diffuse + specular
2. Texture mapping (Fig7)

Boss:

1. Animation (video)
2. Phong shading (Fig8)
3. Texture mapping (Fig8)

Background:

1. Infinite mountain : full-screen quad is rendered without depth test / shift background texture coordinates continuously on the quad.
2. 3D-height-field terrain : use pixel value of a bitmap as the height of terrain. Illusion of infinity is realized through recursively moving of two terrains.

Others:

1. One lighting source
2. Using material

Appendix:

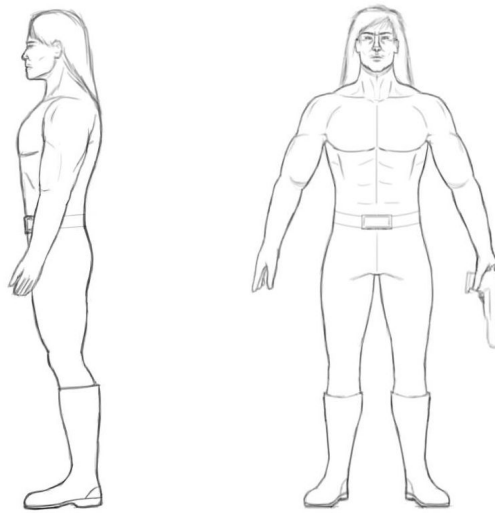


Fig 1. hero sketch

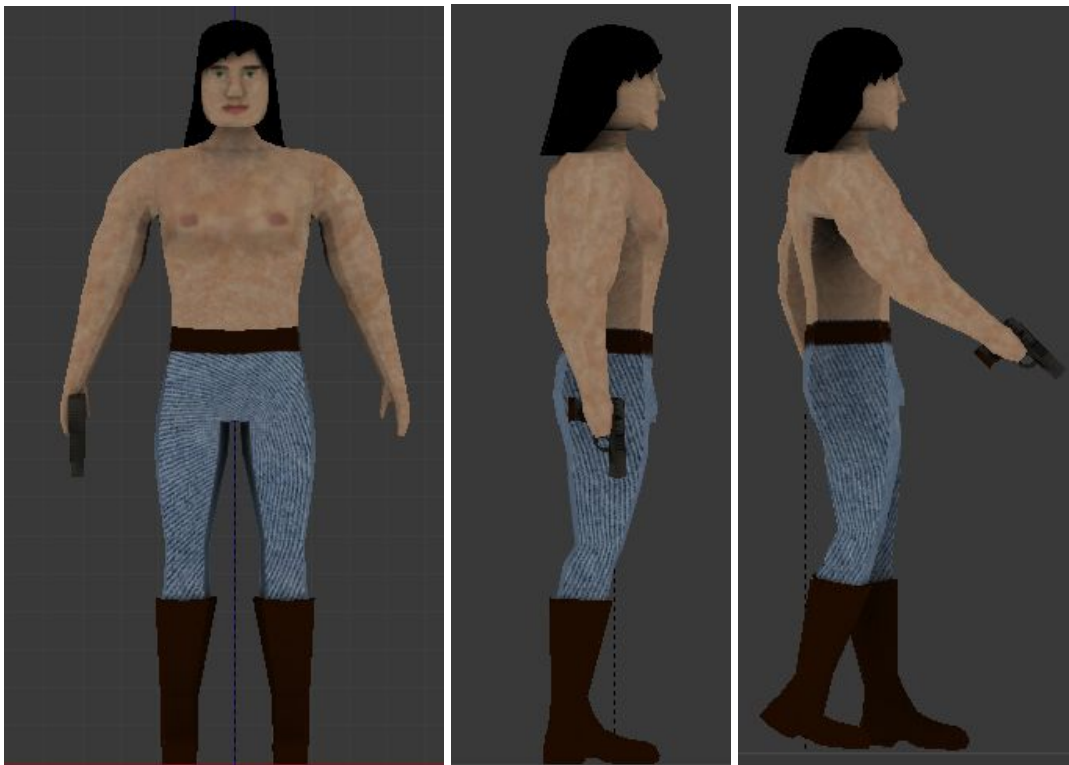


Fig 2. Hero in Blender



Fig 3. Boss Sketch



Fig 4. Boss in Blender

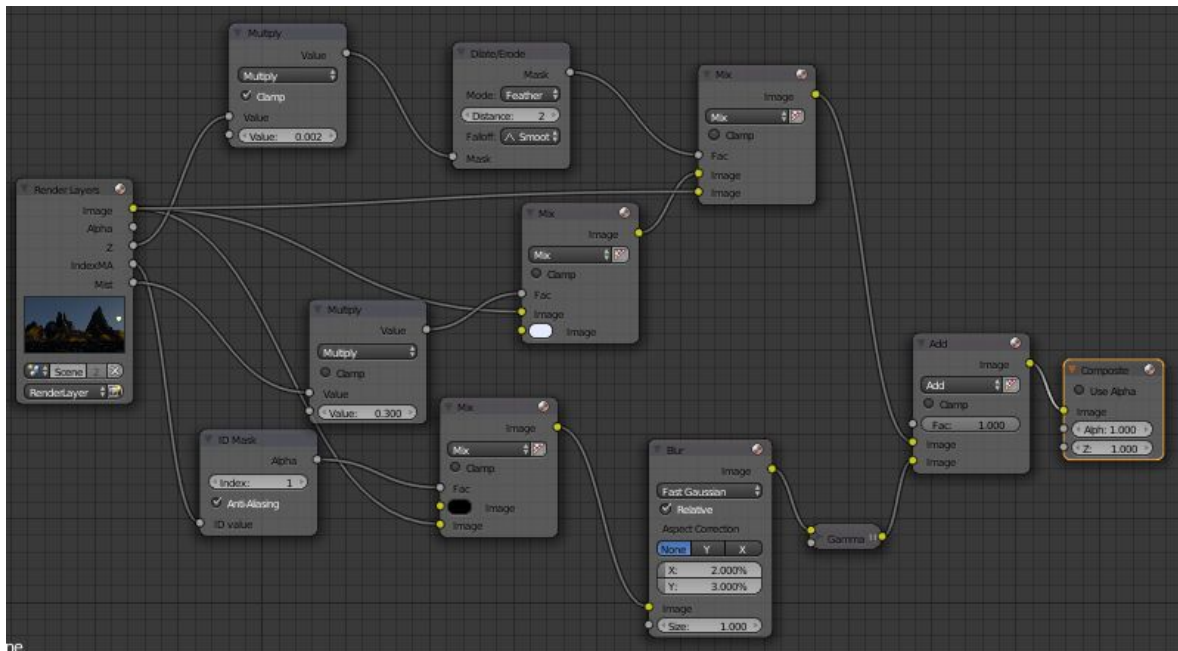


Fig 5. Build Landscape in Blender (cycles render engine)

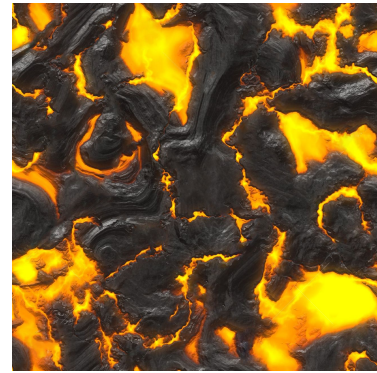
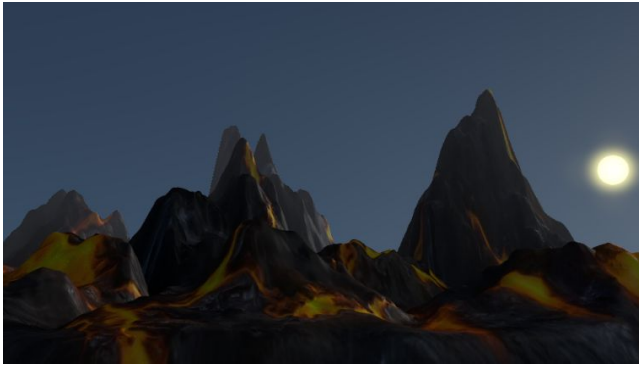


Fig 6. Build Landscape in Blender (texture)

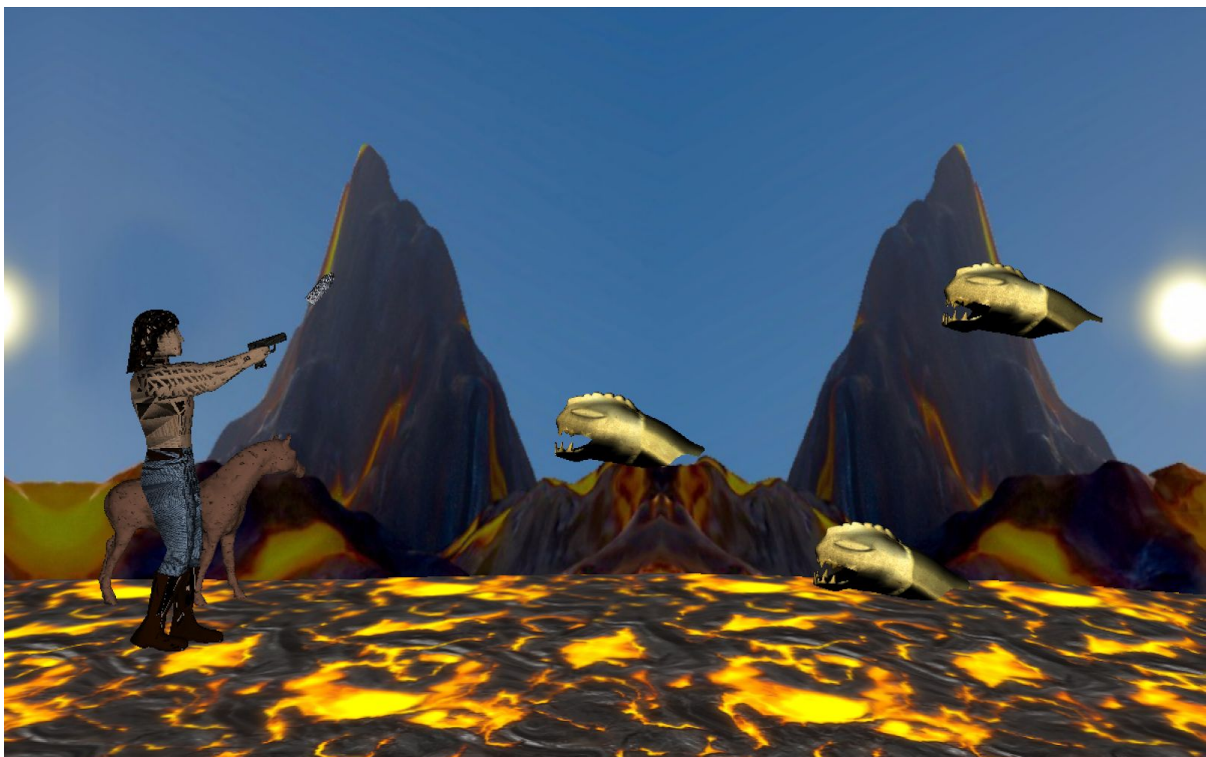


Fig 7. Screenshot

(**hero**: texture mapping, animation, **bullet**: direction control, **background**: infinite mountain, 3D height-field terrain, **enemy**: lighting, material, phong shading)



Fig 8. Screenshot
(**guards**: transformation matrix, **boss**: animation, phong shading)

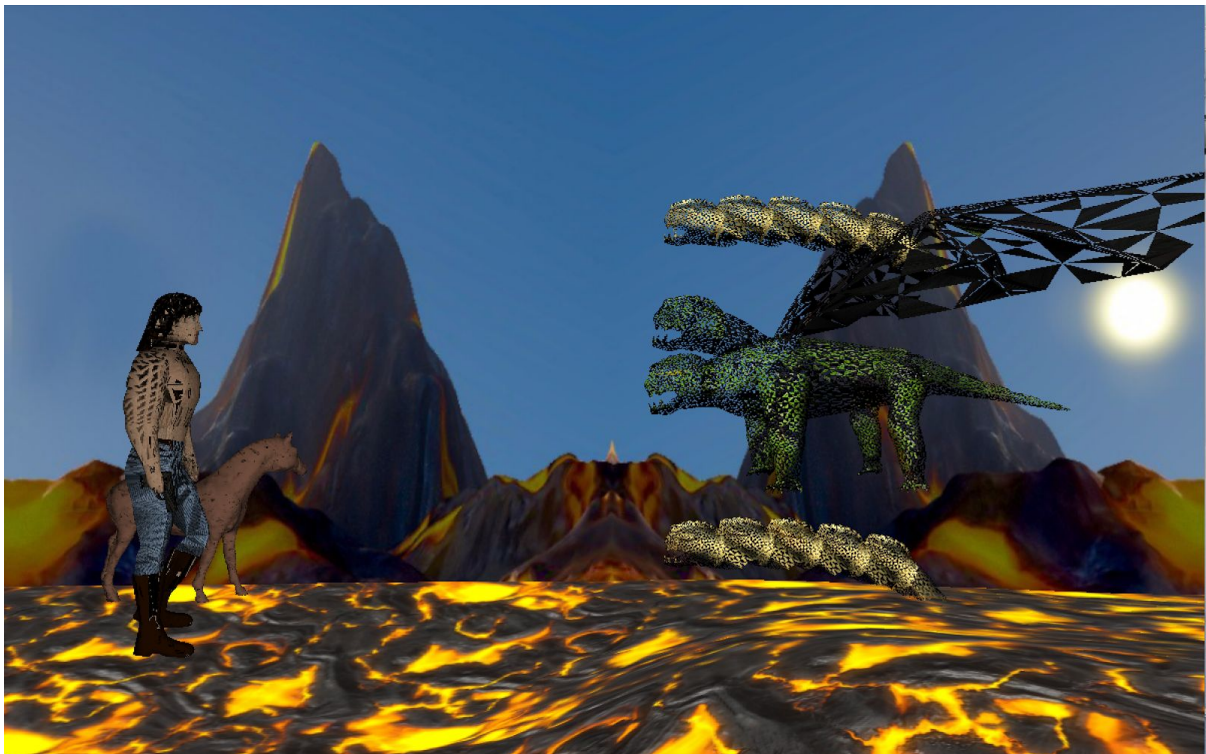


Fig 9. Lose vertices (2)
(**boss**: lose particle vertices)

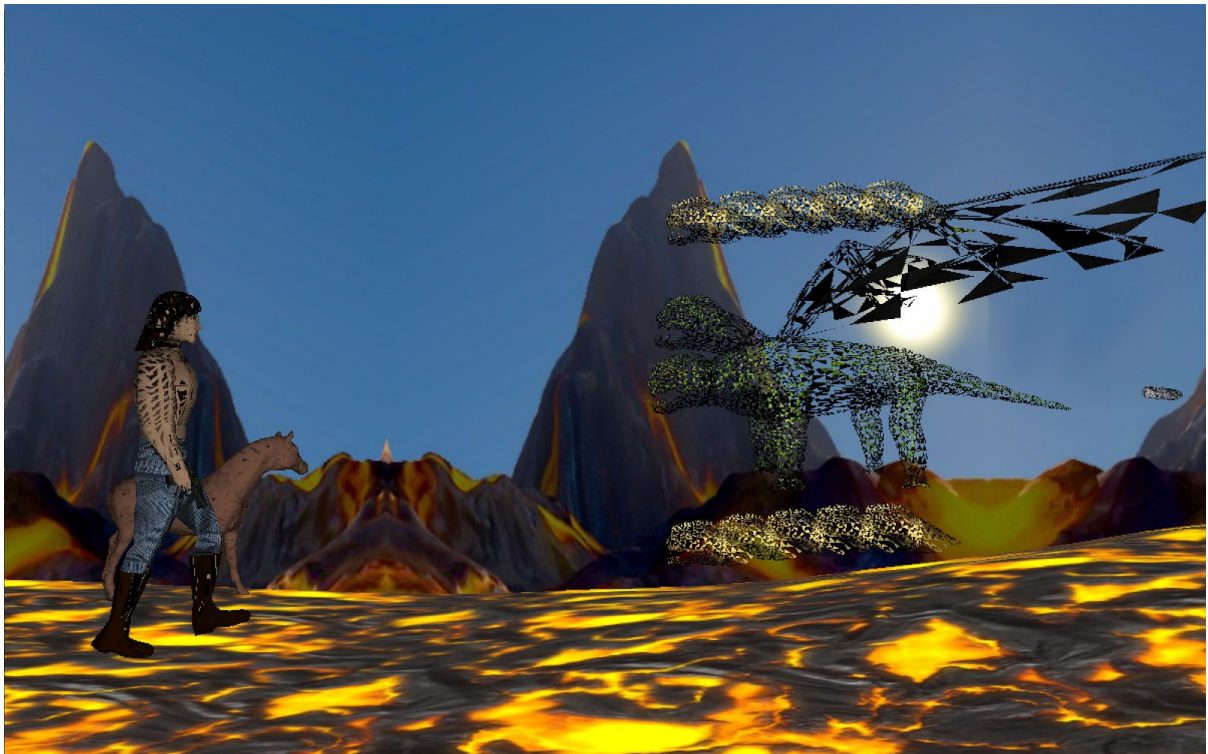


Fig 10. Lose more vertices (3)

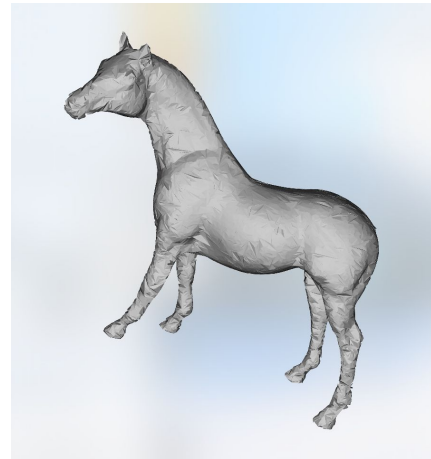
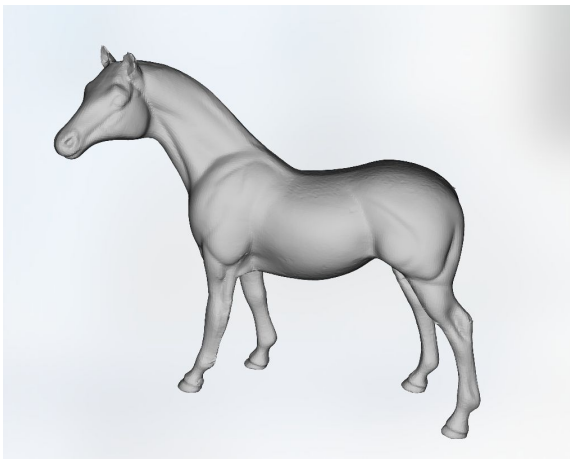


Fig 11. Mesh simplification (from 90000 vertices to around 30000 vertices)