

Realtime-rendering WS2022: Submission 4 (New Year's Eve)

Description of the implementation

Our project is implemented using C++ with OpenGL. It consists of a small town with a couple of houses, lamp posts and surrounding area. Sky is implemented using shadow box.

In the center of our screen there is a rocket, when the main character is close enough to the rocket, a popup inventory menu shows up with a lighter. This menu closes by itself after a few seconds and the rocket is successfully launched. After the rocket reaches a predefined height, it explodes, and particles simulate the following fireworks.

The camera in our implementation is completely independent. Predefined movement consists first of the character walking towards the rocket and launching it. Afterwards there are two other rocket explosions where the camera is positioned automatically in the air so that omnidirectional shadow implementation is nicely visible. After three rocket explosions the demo is completed, and the application is automatically shut down.

If the user decides to take over the control during the demo he can do so by pressing the ENTER button. After that he can freely move around the scene and launch the rockets indefinitely by moving to proximity of the rocket.

Additional libraries

Our project currently uses following libraries

- assimp – used for model loading
- freetype – text rendering
- Glad – OpenGL loading library
- GLFW – OpenGL utilities
- glm – OpenGL mathematics

Hardware

Our implementation works on both our PC's (GTX 760, MX130) as well as on the Vislab's PC with GTX 1060 with stable 60 fps on Fullscreen full HD mode.

Effects

- Phong's lightning model is implemented for array of light sources
- Directional light is implemented as a parallel light rays coming from the infinity in the direction of the location of the moon (tries to mimic light coming from the moon), can be switched on/off with the *D* key
- Spotlight from the camera's perspective (mimics torchlight, *S* key)
- Point light from the moving rocket (*P* key)
- Rocket after explosion emits 100 000 particles and simulates their propagation in the space

- The rocket also acts as a point light source for the omnidirectional shadows (can be nicely seen on poles)

Controls

Camera movement

Our demo is completely independent and does not require any input from the user side. However, if the *ENTER* key is pressed while in the demo, automatic camera movement is disabled, and user is free to move in the scene using *ARROW* keys.

Application Exit

Application can be exited using *ESC* button.

Materials and lights effects

- *Key A*: Ambient lighting On/Off
- *Key P*: Point light from the rocket On/Off
- *Key D*: Directions light On/Off
- *Key S*: Spotlight On/Off

Configuration

Our demo does not contain any configuration file.