

Documentation

Group/Game Name: Grandpa Jones

Brief description of implementation:

The cave system and the first-person character "Grandpa Jones" were created in Blender and imported into the scene using assimp. The cave system includes integrated light sources in the form of torches and one oil lamp at the entrance, which contribute to the overall lighting effects. The man-made entrance and exit area features a brick texture, chosen for its suitability for normal mapping (entrance) and to emphasize its artificial character.

The project features the following implemented effects and techniques:

- ★ Advanced Modelling: CPU-based Particle System
- ★ Texturing: Procedural Texture
- ★ Shading: Simple Normal Mapping
- ★ Post Processing: Bloom/Glow Effect

Additional libraries:

- ★ Physx 4.1
- ★ assimp
- ★ freetype

Gameplay:

Mandatory:

- ★ 3D Gameplay:
 - Implemented assimp-based loader for COLLADA (.dae) files
 - contains complex cave system geometry with >10k polygons
- ★ Playable:
 - Gameplay mechanics include timed objective (reach exit with two collected keys before time expires), physics-based character movement, interactive environment with collision detection
 - Executable builds with proper asset packaging
- ★ Min 60 FPS and Framerate Independence:
 - Achieved through delta-time calculation for movement/physics and timer, batch rendering of static geometry and a bloom effect that uses two-pass Gaussian blur
- ★ Win/Lose Condition:
 - Countdown timer displayed in the top center
 - Victory condition: collect two keys and reach cave exit before timer expires
 - Loss condition: timer expiration
- ★ Intuitive controls:
 - WASD for continuous movement

- Space for jump (event callback)
- Mouse look
- F1 to toggle debug camera
- M to display a map of the cave system
- ★ Intuitive Camera:
 - Primary first-person camera locked to player
 - Secondary debug camera with free movement
- ★ Illumination model:
 - Phong lighting with multiple point lights (torches) and material parameters for ambient, diffuse, and specular reflection
- ★ Textures:
 - Mipmapped textures with trilinear filtering
 - Special implementation:
 - Brick texture with normal map at entrance
 - Procedural cave wall texture using FBM noise
- ★ Moving Objects:
 - Physics-driven player movement
 - Dynamic particle system with 500+particles
- ★ Documentation:
 - Here it is :)
- ★ Adjustable Parameters:
 - Parameters can be adjusted using the ini-File under \assets\settings. Screen Resolution (Width/Height) and Fullscreen-Mode (On/Off) are found in window_jones.ini.

Optional:

- ★ Collision Detection (Basic Physics):
 - The game uses PhysX for robust collision detection between the player and the static cave environment, ensuring the player cannot walk through walls or geometry
- ★ Heads-up Display:
 - A 2D HUD overlay displays a countdown timer, implemented using FreeType text rendering and blending.
 - A 2D map of the cave can be displayed and hidden by tapping M.

Effects:

Lighting:

- ★ Lighting: Lightmap using Separate Textures
We plan to introduce baked lightmap textures for the entrance area to reduce load, but we didn't manage to complete until the submission date.

Advanced Modelling:

- ★ CPU Particle System:
The game has a particle system for showing flames on the torches on the cave walls and also in the player's hand. For this to work we need a base quad that represents

all our particles in the system. We therefore introduced two new shaders that take the instance's position and color. The particles have a randomized starting point around a point on the torch and randomized lifespan. They are also always facing the camera.

<https://www.opengl-tutorial.org/intermediate-tutorials/billboards-particles/particles-instancing/>

Animation: -

Texturing:

★ Procedural Texture:

To visually enhance the cave system, we use a procedural shader based on gradient noise, which generates a rock-like texture for the terrain.

Shading:

★ Simple Normal Mapping:

Simple Normal Mapping is implemented by altering the surface normal using a normal map texture. The normal map is sampled with UV coordinates aligned to the surface, without tangent space transformation, making it suitable for axis-aligned surfaces like quads and boxes. This effect is visible in the entrance area with a brick texture. The effect can be toggled on or off pressing N during runtime.

<https://www.rastertek.com/gl4linux20.html>

<https://learnopengl.com/Advanced-Lighting/Normal-Mapping>

<https://eliemichel.github.io/LearnWebGPU/basic-3d-rendering/lighting-and-material/normal-mapping.html>

Advanced Data Structures: -

Post Processing:

★ Bloom/Glow:

The Bloom effect is implemented as a post-processing step: bright areas of the rendered scene are extracted, blurred using a two-pass Gaussian blur and then added back onto the original scene. This creates a soft glow around bright light sources such as torches and the oil lamp in the entrance area. The strength of the effect is primarily controlled by the exposure parameter, which can be adjusted at runtime. The overall intensity factor for bloom is fixed in the shader code.

<https://learnopengl.com/Advanced-Lighting/Bloom>

<https://www.rastergrid.com/blog/2010/09/efficient-gaussian-blur-with-linear-sampling/>

<https://kalogirou.net/2006/05/20/how-to-do-good-bloom-for-hdr-rendering/>

Other special features:

The game presents a sleek start menu featuring two clear options: "Play" to dive into the adventure, and "Quit" to exit gracefully. Should the timer expire, players are greeted with a Lose screen that offers a straightforward button to return to the main menu, (offering the option to Play or Quit again). Similarly, upon victory, a Win screen provides the same seamless option to navigate back to the menu.

While the current menu setup focuses on simplicity without adjustable settings, future updates could introduce customizable options to enhance player experience.

Walk-through:

Upon starting the game, players are greeted by the main menu, where they can choose to begin their adventure as Grandpa Jones. Once the game starts, players find themselves at the man-made brick entrance of a mysterious cave, illuminated by flickering torches and an oil lamp standing on some sort of trunk? But there's no time to lose!

The cave is already shaking, and you have to find two keys and the exit in time if you don't want to make the cave your grave...

Using classic first-person controls (WASD to move, mouse to look around, space to jump), players explore the atmospheric cave environment. The walls are adorned with procedural textures, and the path is lit by glowing torches, each accompanied by dynamic particle effects that add to the immersive atmosphere.

As time ticks away, players must stay alert and avoid getting lost in the labyrinthine passages. A hand-drawn map of the cave serves as their only guide.

If they reach the exit in time, a Win screen celebrates their success and offers a chance to play again. If the timer expires, a Lose screen appears, allowing them to return to the main menu and try once more.

Throughout the experience, visual effects such as bloom and normal mapping enhance the sense of depth and realism, making each run through the cave a visually engaging adventure.