UE COMPUTERGRAPHIK
186.831

Samy the Salmon - 1st Submission Documentation

Martin Mautner
e1127229@student.tuwien.ac.at

Lisa Kellner
e1428183@student.tuwien.ac.at

Short game description

Samy the Salmon is a skill-based adventure game. The player take control of salmon Samy, helping him reach his birthplace to procreate. Unfortunately, Samy's birthplace is the source of a severely polluted river, making his final journey extremely dangerous and demanding.

Implementation details

Gameplay

At the beginning a start-screen appears where the players can enter their names. By pressing the Enter-key the game starts. Samy can move around in the 3D riverbed freely. The goal of the game is to reach the end of the river with as much health and collected points as possible. The river is polluted with down streaming and rotating stuff which decrease health points by hitting them. There are also rotating coins in the riverbed. Collecting them increases the player's score. At the end of the game, either by reaching the end of the river or dying, a finish-screen with a high score list is shown. The high score is the sum of Samy's health plus the player's score minus the elapsed time of playing the game.

Controls

The player controls Samy with the mouse and the keyboard.

W,A,S,D	Faster, left, slower, right
Mouse	Move camera
Space	Smash forward or jump
Enter	Start game
Esc	End game

Effects

Implemented Effects are:

- **Shadow Maps (with PCF) (1.5 pts**)
  The scene is rendered from the view of a single direction light source (the sun). The green moving firefly light sources are not additionally used. The objects and plants shadow the scene according to the resulting shadow texture. PCF is implemented like in this NVIDIA Whitepaper[1], where a 16 elements long poisson disk is used for the calculations. Furthermore the NVIDIA GPU built-in PCF is also enabled.
- **Water[2] (+ Fresnel-Shading, Normal Mapping) + Reflection + Refraction (2 pts)**
  For the refraction and reflection effect two textures are generated that are sampled when rendering the water surface itself. The refraction texture is generated without the need of an additional pass, by simply rendering to a second color attachment when rendering the scene. The reflection texture is obtained in a second pass, rendering the scene mirrored by the water plane clipping geometry

[1] Integrating Realistic Soft Shadows into Your Game, Source: Engine
http://developer.download.nvidia.com/whitepapers/2008/PCSS_Integration.pdf
[2] Generic Refraction Simulation
http://http.developer.nvidia.com/GPUGems2/gpugems2_chapter19.html

below it (also the backface culling has to be inverted). When rendering the surface both textures are sampled, using a dUdV map to get some distortion. Both values are then combined using Schlick's approximation of the Fresnel factor. Finally, some specular highlights are added based on the water's normal map.

- **Normal Mapping (1 pt)**
  Normal Mapping is implemented, by using a normal map texture to enhance the details of the scene. This effect can especially be seen at the terrain in combination with the green moving light sources. The grass seems to be more detailed and appears more realistic.
- **Fog**
  Some simple exponential pixel fog was implemented to achieve a more realistic look of the underwater scene.

Complex Objects

Non-trivial objects in the game are the plants, coins and the down streaming obstacles like a bear, the Pepsi can, shoes, BB8 etc. The objects have either .obj- or .3ds-formats and are imported with an Assimp loader. The riverbed and the water are self-made and created with blender.

Animated Objects

One sort of the down streaming objects are a pair of boots. The boots are two separated meshes, each boot is one. They are moving together along the riverbed and rotate respectively around each other.

View-Frustum Culling

The green moving firefly light sources (very small boxes) above the water, the down streaming objects and the coins are culled via the View-Frustum. A simple inside-outside test is performed with the Bounding-Boxes of the objects and the View-Frustum. If the objects are not inside, their Scene-Graphs are not rendered.

Transparency

The plants have some transparent parts at the edges of the leaves. A simple alpha-test is used to display them correct.

Key-Mappings

These key-shortcuts toggle some functionalities.

F2	Statistics	Turns statistics overlay on/off.
F3	Wire Frame	Turns Wire Frame move on/off.
F4	Texture-Sampling Quality	Switches between different Texture-Sampling Quality modes. Nearest Neighbor and Biliniear.
F5	Mip-Mapping Quality	Switches between different Mip-Mapping Quality modes. Nearest Neighbor, Linear and off.
F8	View-Frustum Culling	Turns Frustum Culling on/off.
F9	Transparency	Turns transparency on/off.

Illumination and materials

The scene is lit using the Phong illumination model and two light-sources types:

1. A static directional light, simulating the sun
2. Some green point lights spinning over the water, simulating some fireflies

The terrain, water and the objects have all different materials applied to them. Moreover, normal mapping is applied to the terrain giving it more detail. This effect can be especially detected by the moving light sources. To this end, the tangent space is computed automatically by Assimp. The lighting effects can be seen especially well when looking at the obstacles, since they rotate and move downstream at the same time.

Additional libraries

The engine Aardvark[3] from the VRVis Research Center is used to create this game. Assimp[4] is used for loading models from files.

Special Features / Complex interaction sequences

With the Space-key Samy can smash forward or jump out of the water. So the player is also able to collect the coins above the river. If Samy jumps out of the water and strands on the terrain, he randomly makes little jumps until he gets back into the water.
Some coins are placed outside of the water above the terrain (at the first right turn and on the left side before the goal). For reaching them the player has to jump out of the water and / or bounce a little bit at the terrain.
There is also some background music – The Salmon dance from The Chemical Brothers.

[3] https://github.com/vrvis
[4] http://www.assimp.org