Semestrální práce 17
Hierarchical View-Frustum Culling for Z-buffer Rendering

Tomáš Bubeníček[1]
Katedra počítačové grafiky a interakce,
Fakulta elektrotechnická, ČVUT Praha

Abstract

Implement hierarchical view frustum culling for large scale scenes consisting of triangles. First, construct a bounding volume hierarchy (BVH) using top-down method, middle point subdivision. Avoid rendering such BVH nodes that cannot be visible (out of viewing frustum) usually known as view frustum culling.

Keywords: Frustum culling, BVH, GPU rasterization

Done during the Data Structures in Computer Graphics course (B4M39DPG) under Vlastimil Havran. The example scenes use these .obj files provided by him, which are not distributed with the project:

- *asianDragon.obj*
- *block_in_pompeii.high_lod_combined.obj*
- *City4M.obj*
- *part_of_pompeii.01.final.combined.obj*
- *PowerPlantM.obj*

For the scenes to load correctly, the files are required to be put in the data directory. The first loading of the files will take a while and create a binary cache of these files, so subsequent loads will be fast.

For compiling on Linux, requires these development libraries: SDL2, GL, GLEW, glm. Windows dependencies are distributed with the project. Linux dependencies should be available in your distribution repositories.

Author and contact: Tomáš Bubeníček, `bubentom@fel.cvut.cz` or `tombuben@gmail.com`

Compiled with Visual Studio 2017 using Intel C++ Compiler 19.
Usage: `FrustumGL.exe -scene [path to scene]`

Reference

[AM00] Ulf Assarsson and Tomas Moller. Optimized view frustum culling algorithms for bounding boxes. *Journal of graphics tools*, 5(1):9–22, 2000.

[1] B4M39DPG – Tomáš Bubeníček, zimní semestr 2019/20

[GH01] Gil Gribb and Klaus Hartmann. Fast extraction of viewing frustum planes from the world-view-projection matrix. *Online document*, 2001.

[SJ02] Daniel Sýkora and Josef Jelínek. Efficient view frustum culling. In *Central European Seminar on Computer Graphics*, 2002.