Berner Fachhochschule
Haute école spécialisée bernoise
Bern University of Applied Sciences

3D Terrain with Level of Detail

Written report for the module
BTI3041 – Project 2
by

Amar Tabakovic

Bern University of Applied Sciences
Engineering and Information Technology
Computer Perception & Virtual Reality Lab

Supervisor
Prof. Marcus Hudritsch

October 4, 2023

Abstract

TBD.

Contents

Chapter 1

Introduction

In the field of 3D computer graphics, rendering is one of the central tasks. Many practical applications of 3D computer graphics make use of terrains, such as flight simulators, open-world video games, and Geographic Information Systems (GIS) [1, p. 185]. At the same time, rendering large and constantly visible objects, such as the terrain, is computationally expensive and optimizations are necessary in order to avoid performance deficiencies.

One area which offers potential for optimizations is the *level of detail (LOD)* of objects. The concept of LOD is based on the idea that the farther away an object is, the fewer details are going to be visible to the human eye.

The problem of rendering terrains spawned numerous algorithms and approaches specifically for this purpose.

1.1 Goals of this Project

The main goal of this project is to gain an overview over the field of terrain LOD. This includes analyzing and comparing a selection of terrain LOD algorithms and approaches, including implementing as part of a demo computer application.

1.2 Structure of the Report

This report is structured as follows:

-

Chapter 2

Existing Work and Literature

This chapter gives an overview of the current state of terrain LOD research and lists the main literature and resources used in this projet.

LOD and terrain rendering are areas which have been researched heavily.

2.1 Research Articles and Publications

Test

2.2 Level of Detail for Computer Graphics

Level of Detail for Computer Graphics by Luebke et al. [1] is a reference book for LOD rendering published in 2002. The book builds on top of years of research in the area of LOD and provides an overview to many LOD techniques. For this project, chapter 7 "Terrain Level of Detail" of the book is especially relevant, as it dives into the topic of LOD for terrains specifically. It covers basic approaches and techniques for terrain LOD, common problems that can occur during rendering of terrains and their solutions, and a catalog of terrain LOD algorithms.

2.3 Virtual Terrain Project

The *Virtual Terrain Project* [2] was a project run from 2001 to 2013 that consisted of a collection of software, information and resources on terrain modelling and rendering.

Chapter 3

Terrain LOD in Real-world Systems

This chapter lists a selection of real-world examples of terrain LOD algorithms in use, such as game engines and geographic information systems (GIS).

3.1 Game Engines

3.1.1 Godot

Godot is a cross-platform game engine written in C#, C++ and its own scripting language GDScript. Terrains are supported in form of extensions developed by community members, which can be installed and used in Godot projects by game developers. One such extension is Terrain3D by Cory Petkovsek [3] written in C++ for Godot 4. The LOD approach used in this extension is based on Geometry Clipmaps by Hoppe and Losasso [4]. The concrete implementation of the geometry clipmap mesh code was created by Mike J Savage [5].

3.1.2 Unity

3.1.3 Unreal Engine

Unreal Engine is another cross-platform game engine written in C++ and features an integrated terrain system (called the Landscape system). The technical documentation of Unreal Engine 5 mentions utilising geomipmapping for handling LOD for landscapes [6]. Geomipmapping is a terrain LOD approach developed in 2000 by de Boer [7].

3.2 Geographic Information Systems

Chapter 4

Results

Chapter 5

Discussion and Conclusion

5.1 Further Work

5.2 Reflexion

Bibliography

[1] David Luebke, Benjamin Watson, Jonathan D. Cohen, Martin Reddy, and Amitabh Varshney. *Level of Detail for 3D Graphics*. Elsevier Science Inc., USA, 2002.

[2] Virtual terrain project. `https://vterrain.org`.

[3] Cory Petkovsek. Terrain3d. `https://github.com/TokisanGames/Terrain3D`, 2023.

[4] Frank Losasso and Hugues Hoppe. Geometry clipmaps: Terrain rendering using nested regular grids. *ACM Trans. Graph.*, 23(3), 2004. `doi:10.1145/1015706.1015799`.

[5] Mike J Savage. Geometry clipmaps: simple terrain rendering with level of detail. `https://mikejsavage.co.uk/blog/geometry-clipmaps.html`, 2017.

[6] Epic Games. Landscape overview — unreal engine 5.3 documentation. `https://docs.unrealengine.com/5.3/en-US/landscape-overview/`.

[7] Willem H. de Boer. Fast terrain rendering using geometrical mipmapping. In *The Web Conference*, 2000. URL: `https://api.semanticscholar.org/CorpusID:7296927`.