Terrain Generation

Tiju Thomas

CIS 565 – GPU Programming Final Project
Final Presentation

14th December 2012

Outline

- Introduction
- Goal
- Details
- Demo

Introduction

- Inspired by my love for the outdoors
- Terragen (http://planetside.co.uk)

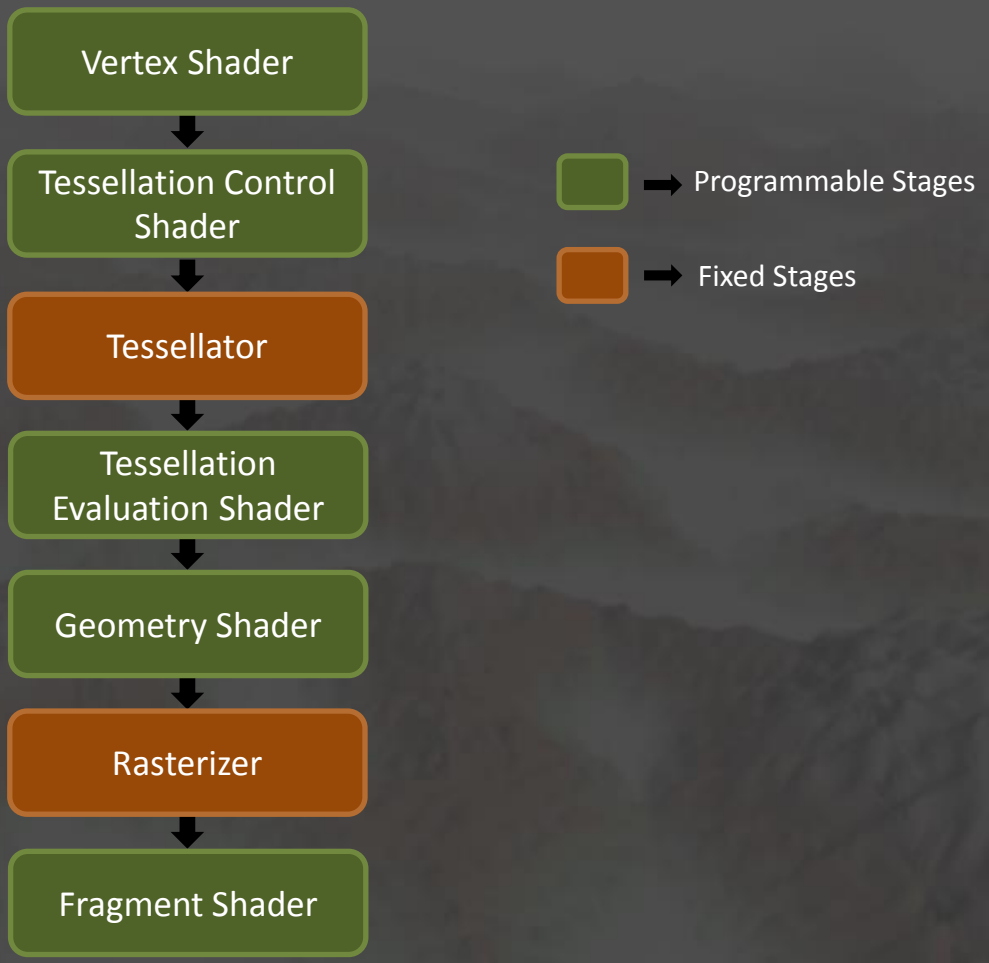
Goal

- Create a basic Terrain Generator
- Tessellation based on distance from camera
- Different terrains based on noise (and maybe height maps)

Implementation Details

- Flat Grid
- Height using Perlin Noise
- Displacement in Vertex Shader
- Tessellation Shaders
- Face normals using Geometry shader
- Color assigned using Fragment Shader
- Multiple rendering passes for screen space effects

Pipeline


Vertex Shader
Tessellation Control Shader
Tessellator
Tessellation Evaluation Shader
Geometry Shader
Rasterizer
Fragment Shader
Programmable Stages
Fixed Stages

Demo

Performance

- Terrain Size – 1600 x 2500

	Non-Tessellated	Tessellated	Smart Tessellation
Triangles	160K	160K*400	~20K*400
Mesh*	21 fps	1.6 fps	7 fps
Shaded*	11 fps	1.8 fps	8.5 fps

* Numbers based on output size 1024 x 1024

Future

- Smoother Normals
- Texturing
- Controllable Deformable Terrain

References

- OpenGL Insights book
- Texturing And Modeling – A Procedural Approach book
- Perlin Noise (http://www.sci.utah.edu/~leenak/IndStudy_reportfall/Perlin Noise on GPU.html)
- The Little Grasshopper blog (http://prideout.net/blog/?p=48)

Thank you

Blog: http://gputerrain.blogspot.com
Code: https://github.com/tijutv/GPU-Terrain-Generation