Progress Report

Bruce Alcock

A Procedural, Minimal Input, Natural Terrain Plug-in for Blender

Supervisors: Kevin Glass and Shaun Bangay

29 March 2007

1 Previous Short Term Objectives:

1.1 Website

Include links to references, the project proposal in both textual and PowerPoint formats, and weekly progress reports.

1.2 Blender

Set up the scene in an efficient manner, and create more realistic terrain, by figuring out how the Fractional Brownian Motion (fBm) generator works.

2 Progress:

2.1 Website

The project proposal in both textual and PowerPoint formats, weekly progress reports as well as screenshots have been included on the project's research website, and with the help of some PHP scripting it is now exceedingly trivial to add blog comments, progress reports and screenshots.

It was decided that placing links to the project's references was unnecessary, and also difficult since more reading is being done. It may also infringe copyright issues to publicly share research documents without the author's permission.

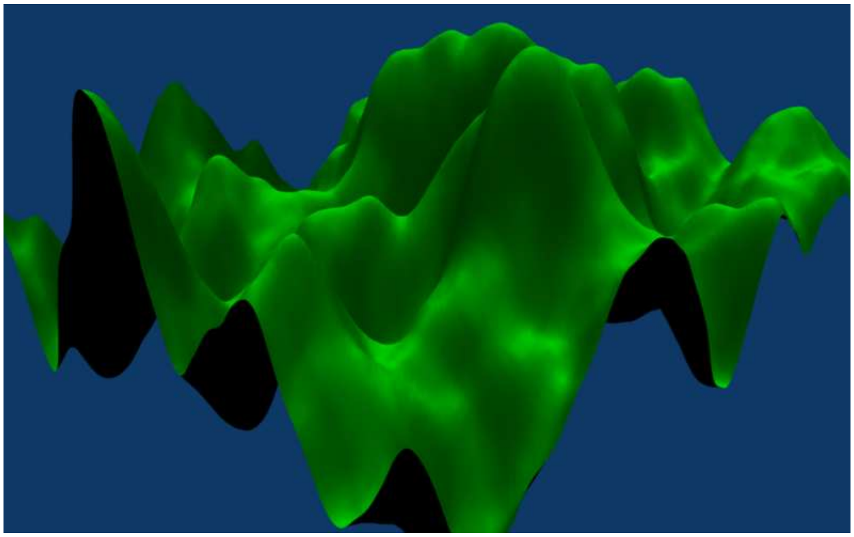

Figure 1: Terrain rendering

2.2 Blender

The script now removes the standard cube and lamp that are visible when Blender starts if they're present. It sets a new camera position which makes for standard screenshots, and a lamp above the terrain, which has been changed to a sun light source. There are also 3 levels of standard Catmull and Clark subdivision applied to the mesh, but this is not shown during realtime so as to enable faster realtime manipulation.

The fBm generator method seems to have been better understood as is visible along with all the other updates in Figure 1. The terrain being produced is much more realistic than the previous renderings, and most importantly Figure 1 was generated simply by running the script and selecting the render option in Blender, which means successive screenshots and changes are now easy to compare because the scene is set up in a standard way.

2.3 Literature

Another paper [1] was read: it proposes a system of creating ridges by dispatching ridge particles randomly across the map, with an altitude and direction and these then form ridges along the terrain, decreasing in height over distance and subject to fBm collisions with other ridges. The paper also proposes a similar idea for creating rivers, by randomly dispatching river particles at the top of the ridges: these are balls which have mass and are subject to gravity, effectively carving out rivers of the terrain. This could possibly be implemented along with the fBm terrain generation of [2] and afterward erosion [2] applied to give a more realistic appearance. The river generation is superior to the approach used in [3] of using Squig curves, because this is based on a (albeit simple) physical model.

Other papers mainly related to Level of Detail (LOD) have been obtained and will be analyzed.

3 Problems:

None

4 Objectives For The Next Two Weeks:

- Create a method to erode the terrain as detailed in [2] in the script.
- Create a way of specifying areas to have flatter ground.
- Create a method to roll river particles down the terrain to create river systems as detailed in [1].

References

[1] Fares Belhadj and Pierre Audibert. Modeling landscapes with ridges and rivers: bottom up approach. In *GRAPHITE '05: Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 447–450, New York, NY, USA, 2005. ACM Press.

[2] F. K. Musgrave, C. E. Kolb, and R. S. Mace. The synthesis and rendering of eroded fractal terrains. In *Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 41–50. ACM Press, 1989.

[3] Przemyslaw Prusinkiewicz and Mark Hammel. A fractal model of mountains with rivers. In *Proceeding of Graphics Interface '93*, pages 174–180, Toronto, Ontario, May 1993.