NuGenCMapSpatial

The aim of the project is to take 2D image data sources and produce displacement mapped geometry and render this in a 3D context.

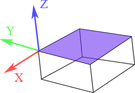
All work is done on behalf of and is the property of Genetibase Inc.

# Core Features

* Rendering height-maps
  + Full 3D view with camera support etc.
* Pure geometry based height-maps
  + Viewport clipping via quad trees
* GPU based rendering of height-maps
  + Simple height rendering from texture
  + Geometry clip-maps for large/detailed images/scenes
* Height-map sampling options
* Diffuse-map generation from height-map
* Height-map interpretation options / profiles
* Various overlay options
* Cross-section support
* Lighting & shading model

# Feature Design

### 2.1 Height-Map Rendering

Once a data source and profile have been specified and the data processed into geometry (and possibly shaders too) a height-map can be rendered in a 3D view. The view should have camera options and input.

The geometry itself will be composed of a 2D tessellated plane (XY) and a Z axis which represents the heights of the height-map. Sides to the rendering – to make it look like a chunk cut out of something bigger – would be optional, along with a flat or parallel bottom surface.

### 2.2 CPU based geometry calculation

This will be the basic configuration that can be run on pretty much any machine. It will also be the slowest on high-end machines.

It relies on the geometry vertices to be recalculated every time the height-map changes or the level of detail changes. This is done using a software CPU height-map sampler.

### 2.3 GPU based geometry calculation

This configuration required a Shader Model 3.0 compliant graphics card, in which the Z axis values are added onto each vertex from runtime texture lookups, all on the GPU. This allows very quick geometry setup, allowing for very quick level of detail changes.

### 2.4 Height-Map Sampling Options

Sampling options for the height-map are required to finely adjust the final geometry output. This basically entails the number and pattern of samples taken for each geometry vertex. This obviously effects the time taken to sample each point and therefore the time taken to generate the geometry (or render), but also how accurate the geometry representation is.

#### 2.4.1 Planned Sample Patterns

The main point of sampling patterns is to get a better approximation to the area of height values the resultant sample represents.

Typically a sample result represents a range of +0.5 (samples) to -0.5 on both X & Y axis, this forms a square of size 1 sample centered on the main sample point.

1 Sample is fairly self explanatory –

4 samples consists of 1 sample either side of the point on the both axis -

5 samples are the same as 4 but include the central point.

#### 2.4.2 GPU Optimizations

As we don’t want to have to sample many texture lookups per vertex on the GPU (as they are comparatively expensive) we can simply render a new texture for each level of detail, which includes the sum averages for each sampling point at each pixel using the exact correct texture size.

### 2.5 Cross-Section Support

Cross-sections can be taken from the 3D view by drawing out a plane; this then produces either a graph based on the rendered/existing geometry or purely from height-map sampling. This graph of points can then be rendered in 2D or as a 3D overlay.

### 2.6 Overlays

Overlays can be true 3D geometry such as other rendered height-maps, 2D texture projections, UI graphics or helper geometry.

### 2.7 Height-map Profiles

Profiles are required in order to tell the program how to interpret colour values in a given image / data-source. Reading of various geospatial formats will be done through GDAL (Geospatial Data Abstraction Layer).

Profiles should be available for the following purposes

* Simple height-map greyscale (8-bit, 16-bit 24-bit single values etc.)
* RGB colour bands
* Terrestrial satellite imagery
  + Digital Elevation Models (DEM)
    - GTOPO30
    - SRTM
    - SRTM30Plus
* Medical scan imagery
  + MRI slices

### 2.8 Lighting & Shading Model

Lighting and shading should be dependant on profile in use and options selected etc.

Geometry will be lit with aid of a normal-map, which can be specified explicitly or generated from the height-map. Lighting will be either plain per-vertex, per-pixel, normal-mapped; perspective shadow maps may also be used.

#### 2.8.1 Generating Normal-Maps from Height-Maps

Generating this map is very similar to generating per-vertex normals from per-face normals. We essentially sample each pixel, measuring the normal vector using the surrounding pixels, then averaging the values (all internal pixels have 4 sources, external have only 2, corners only 1 which saves keeping tabs of source multiplicities per-pixel). A GPU version of this could be devised for some speed increase on supported hardware.

### 2.9 Diffuse-map generation from height-map

Diffuse-maps are essentially used for the texture we want to give the final geometry, this can be typically the source image but in most cases we need to generate some colour data from the height data.

#### 2.9.1 Height Bands

This is a general principle to assign locations of colour in the range of 0.0 – 1.0 which can then be sampled at arbitrary values within that range to get a colour representative of that height (value) within the context of this colour band scheme.

This mechanism can then be used in conjunction with profiles to give typical colour band schemes for each profile type.

### 2.10 Data Management

A Data Manager is required to turn large data sources into subsets of varying size/detail, this includes both geometry quads/planes as well as images/textures. This depends on what the hardware can support in part for texture sizes, but textures themselves should handle their own sub-data via mip-mapping. This should fit nicely into a quad-tree for caching and lookups etc. This may also include tiling from mosaic data sources, seamlessly.

Caching can be output to disk to save re-sampling in most cases when a resource outlives memory usefulness.

DataSource

DataManager

Dataset

# Bibliography

Geo-Innovations. (n.d.). Retrieved from http://www.geoinnovations.co.uk/html/Relief.html

*GIS*. (n.d.). Retrieved from http://ioc.unesco.org/oceanteacher/OceanTeacher2/02\_InfTchSciCmm/05\_progagen&orgs/esip/gisproducts&services.htm#glcf

*GTOPO30 documentation*. (n.d.). Retrieved from http://www.eorc.nasda.go.jp/JERS-1/GFMP/AM-3/docs/html/gtopo30.htm

*The Shuttle Radar Topography Mission*. (n.d.). Retrieved from http://www.personal.psu.edu/mem418/SRTM\_Project3C.htm

Wikipedia. (n.d.). *Digital Elevation Model*. Retrieved from http://en.wikipedia.org/wiki/Digital\_elevation\_model

Wikipedia. (n.d.). *Displacement mapping*. Retrieved from Displacement mapping: http://en.wikipedia.org/wiki/Displacement\_mapping