# A Heightfield on an Isometric Grid

Morgan McGuire*
Brown University

Peter G. Sibley†
Brown University

## Abstract

We describe a data structure for improved quality of real-time rendering for large and dynamic heightfields on programmable graphics hardware. The new *isometric heightfield* has a more accurate surface and smoother shading, normals, and curvature.

**CR Categories:** I.3.3 [Picture/Image Generation]: Display Algorithms; I.3.7 [Three-Dimensional Graphics and Realism]: Color, shading, shadowing, and texture

**Keywords:** heightfield, terrain

## 1 Motivation

Despite good *irregular* tessellations methods like ROAM, heightfields remain popular for games and scientific visualizations because the regular grid allows efficient on-disk storage and lends itself well to dynamic data like simulated ocean waves and deformable terrain. Regular tessellations work well in hardware: triangle stripping is easy, and subsampling provides level-of-detail (LOD) control; the fixed memory footprint simplifies vertex buffer management and paging of subsections (tiles) from disk.

A conventional heightfield tessellates a surface by splitting regular grid squares along a consistent diagonal, forming right triangles in the *xz*-plane. Surfaces whose larger principle curvature is along the diagonal end up badly represented, as shown by the shaded mesh and Gaussian curvature in Figure 1, giving a misleading impression of roughness. The isometric heightfield ameliorates these phenomena at almost no cost.

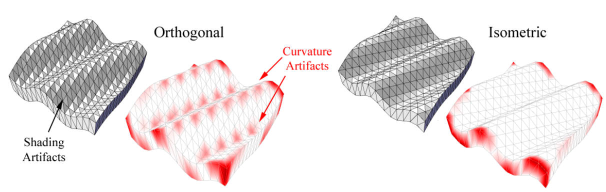


Figure 1: The isometric heightfield minimizes the curvature and shading artifacts that make an orthogonal heightfield appear rough.

## 2 Data Structure

The isometric heightfield is built on a grid with 60°-axes that produce an equilateral tessellation, which is maximally symmetric. Musgrave [1998] proposed creating an isometric heightfield for raytracing by shearing an orthogonal heightfield into a parallelogram.

*e-mail: morgan@cs.brown.edu
†e-mail: pgs@cs.brown.edu

We propose an alternative structure (Figure 2) that slides odd rows horizontally and adds isolateral triangles to even rows. This creates square tiles, which are necessary for mapping square textures without distortion and are more practical for interacting with existing square-grid infrastructure.

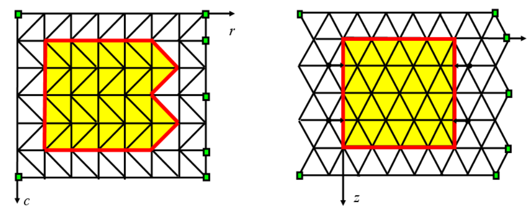


Figure 2: (a) Grid Space (b) World Space

We show how to sample elevations between vertices (necessary for determining locations on the surface) and derive isometric equivalents of other useful heightfield algorithms: creating triangle strips, subsampled LOD, fast shadow volumes, and an isometric version of Shankel's [2002] fast computation for vertex normals.

Each vertex in video memory is compressed to two bytes and then reconstructed during rendering by the vertex processor. This allows large data sets in video memory and efficient streaming of dynamic and paged-in data like those in Figure 3. The top image is a false-color visualization of a region of Mars created from NASA elevation data, created in collaboration with James Head III of the Brown Planetary Geology department. There are 250M triangles in the data set, which have been divided into 2048 tiles. The bottom shows Wake Island from the game Battlefield 1942, used with permission of Digital Illusions. Not shown in the figure, 100 animated characters simulate the additional rendering demands of a real game. Both data sets render at 45fps on consumer hardware.

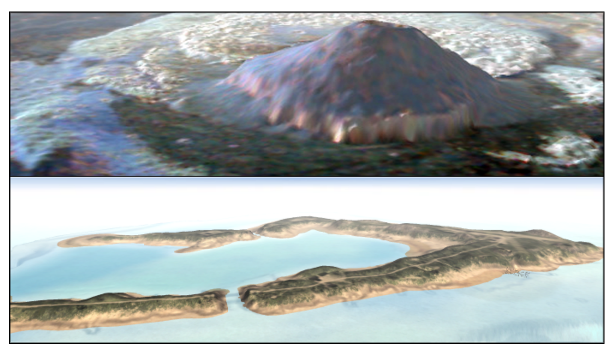

Figure 3: Olympus Mons, Mars and Wake Island, Earth

We measured the elevation, normal, and shading error error from a heightfield approximation of a surface. The isometric error rate averages 25% less than that of a conventional heightfield and is strictly lower independent of feature orientation and frequency.

## References

MUSGRAVE, F. K. 1998. Grid tracing: Fast ray tracing for height fields. Tech. Rep. YALEU/DCS/RR-639, Yale Dept. of Computer Science, July.

SHANKEL, J. 2002. Fast heightfield normal calculation. *in Game Programming Gems 3, Charles River Media, Inc.*, 344–348.