Terrain Representation Using an Orientation Data Model
TERRAIN REPRESENTATION

Foreshadowing:

- **Hand-Drawn relief shading:** time-consuming and expensive
- **Analytic methods:** faster, consistent, reproducible results
- **Often, the only data input is a DEM**
- **DEM may not be the most practical data model for terrain representation workflow**
- **Changing the data storage model:**
  - Reduces run time for many analyses
  - Opens the door to additional techniques

**Examples**
ANALYTIC
TERRAIN
REPRESENTATION

THIS EXAMPLE
COMBINES HEIGHT &
ORIENTATION DATA

TWO COMPONENTS:

**Elevation** = Tint

**Orientation** = Shade
HEIGHT / ELEVATION
DEM is symbolized directly – Easy and Fast
ORIENTATION

DERIVED FROM THE DEM USING A LENGTHY INTERNAL PROCESS

\[
\sin(\theta)^2 + \cos(\theta)^2 = 1
\]

Lots of Math
RELIEF SHADING
A.K.A. "Hillshading"

$S \cdot L = \cos(Slope)$

$BV$ (Brightness Value)

$s_x, s_y, s_z$

Surface Orientation Data

$X = \sin(Aspect) \sin(Slope)$

$Y = \cos(Aspect) \sin(Slope)$

$Z = \cos(Slope)$

$s_x, s_y, s_z$

$BV$ (Brightness Value)

$L_x, L_y, L_z$

Interprets orientation from elevation

Slope | Aspect | Trigonometry | Dot Product

Clock icon
RELIEF SHADING
a.k.a. "Hillshading"

S • L = \cos(\text{Slope})

BV (Brightness Value)

X = \sin(\text{Aspect}) \sin(\text{Slope})
Y = \cos(\text{Aspect}) \sin(\text{Slope})
Z = \cos(\text{Slope})

\overrightarrow{S} \cdot \overrightarrow{L} = S_x L_x + S_y L_y + S_z L_z

DEM
Slope
Aspect

Trigonometry
Dot Product
Reduces computational cost of hillshades and the workflows that depend on them.

Surface Normal Vectors can help us do other cool stuff borrowed from the computer graphics world.
Calculating orientation internally from the DEM is expensive.

The intermediate data layers are discarded when hillshade completes.

Memory requirements are higher.

For single-pass hillshades, this is not a big deal.

For multi-directional hillshades, the cost is substantial.
MULT-DIRECTION HILLSHADES

A USEFUL CASE STUDY TO EXAMINE ORIENTATION-SENSITIVE SYMBOLOGY
**Some Multi-directional Methods**

- **MDO**
  - Multi-Directional Oblique Weighted
  - Combines four traditional hillshades obtained from light sources at low altitude.

- **Multi-Directional (Esri)**

- **Sky Illumination Model**
  - The sky is partitioned into many zones, each of which carries an illumination source. Theoretically similar to MDO, but with hundreds of light sources.

---

**Number of Illumination Sources**

- **One**
  - Traditional / Lambert
    - Relief shading technique based on a single light source and the assumption of a Lambertian surface. Shadows & specular reflections are ignored.

- **Many**
  - Multi-Directional Clustered
    - Dynamic technique using a variable illumination vector. Direction is tied to aspect; the terrain is divided into zones of similar aspect with a different illumination source applied to each zone.
A height data source is ill-suited to methods symbolizing orientation.
SURFACE ORIENTATION DATA
HOW TO REPRESENT ORIENTATION AS A DATASET

DEM → Slope → X = sin(Aspect) sin(Slope) → S_x → Saved To Disk

DEM → Aspect → Y = cos(Aspect) sin(Slope) → S_y

DEM → Slope → Z = cos(Slope) → S_z
Surface Normal Vectors are perpendicular to the local tangent.

A 2D example: A line tangent to the surface generalizes its slope at that point. The vector perpendicular to that line encodes the orientation of the surface.

In 3D, the surface tangent is a plane, which captures slope and aspect. The normal vector is perpendicular to that plane.

3D vectors have three components: X, Y, & Z "Normalized" to length one.
\[ \sqrt{2} \frac{dx}{dz} \]

Lots of Math

Surface Normal Vectors
Surface Normal Vectors
Crater Lake, OR
The brightness value is

\[ BV = |S| |L| \cos(\theta) \]

These vectors are normalized to length one:

\[ BV = \cos(\theta) \]

\( \cos(\theta) \) is the dot product of these two vectors.

\[ BV = \cos(\theta) \\
= S \cdot L \\
= S_x L_x + S_y L_y + S_z L_z \]
WORKFLOW:

SURFACE NORMAL & SHADER
NEW (TO GIS) METHODS:

SURFACE NORMAL FILTERING

- Generalization via low-pass filters
- Sharpen edges with a sharpening filter
- "Bump Mapping"
SURFACE NORMAL FILTERING
Describe and implement a data model comparable to other industries.

Code GP tools for ArcGIS Pro to use data in this form. Create simplified versions of selected terrain representation methods:

- MDOW
- Sky Illumination
- Illuminated Contours

Code new GP tools to implement CG effects within ArcGIS Pro:

- Bump Mapping
- Specular Highlights

Suggest future uses of a vector data model for cartography.
METHODS
Storing as a multi-band raster ensures consistency with:
- Coordinate system
- Projection
- Cell size
- Snap / alignment

The 'CraterLake' Raster Dataset contains 4 bands, named for their content.
ArcGIS Pro Python Toolboxes

Relief Shading
• Data conversion tool
• Re-implement the stock hillshade algorithm
• Recreate multi-directional method chains (MDOW, Sky Illumination, etc)

Extra Tools — Other uses for vector data
• 3D graphics effects.
• Analyses involving gradient:
  • Line Integral Convolution
  • Hydrology
Speed is easily measured within the ArcGIS environment using programming tools.

Memory consumption is also measurable from the programming environment.

Compare vs the published implementations in Esri’s TerrainTools.
Bump Mapping: Gives the illusion of texture on a surface.

Specular Highlighting: Identifies areas of spot reflection.
BUMP MAPPING EXAMPLE OUTPUT

Simple Geometry + Bump Map

The bump map defines how the original surface vectors are to be adjusted or perturbed.

The shader works on these adjusted normals, not the surface, so the effect is as if the surface itself were perturbed.

The underlying surface is not modified.
Treats the surface as if it is glossy or reflective.

Highlights are independent of shading. Depend on light source and camera location.

CG METHODS

SCARY
LINEAR ALGEBRA

COMPLEX BUT WELL-MAPPED PROCEDURES
EXPECTED RESULTS & DELIVERABLES
Deliverable: Python toolbox implementing orientation-aware tools.

- Time trials will show speed improvements over currently published tools.
- Speed improvements proportional to the number of constituents in a multi-directional method.
DELIVERABLE:

NEW TOOL:
BUMP MAPPING
"Wave" bump map masked to area of water surface only

BUMP MAPPING TOOL:

A MORE PRACTICAL EXAMPLE
This example uses a false color highlight to emphasize placement.
PROGRESS & MILESTONES

- **Nov**
  - Submit Abstract To Esri UC

- **Dec**
  - 596A Presentation

- **Jan**
  - Functional Testing / Algorithm Proof of Correctness

- **Feb**
  - Code ArcGIS Toolbox

- **Mar**
  - Speed Trials & Performance Testing
  - User Testing / System Testing

- **Apr**
  - Documentation & Examples

- **May**
  - Assemble Results

- **Jun**

- **Jul**
  - Esri UC July 11, 2021

- **596A Presentation**

- **Functional Testing / Algorithm Proof of Correctness**

- **Code ArcGIS Toolbox**

- **User Testing / System Testing**

- **Assemble Results**
THANK YOU

QUESTIONS & COMMENTS: gz5142@psu.edu