Terrain Analysis:

Using slope algorithms

GEO 327G: GIS & GPS
APPLICATIONS IN EARTH SCIENCE

Katelynn Whittington

December 6, 2018

Table of Contents

Introduction

The default algorithm used for the slope geoprocessing tool in ArcMap is Horn's algorithm. With knowing this information, one can assume that Horn's Algorithm surpasses all, but maybe that's not the case, since according to Kevin H. Jones, Fleming and Hoffer's method comes in at a close second. Depending on the type of terrain and how rigid or smooth (any noise occurring), the best method for calculating slope may change.

Per type of terrain, which slope algorithm would be the best to calculate terrain with? And what are the effects of the other methods? Rigid? Smooth?

In order to test this problem, there will be a need for the focal statistics tool and the raster calculator. Detailed explanation of testing methods, found in step 3.

Step 1: Data Collection

The first thing to do is to collect all the data needed for the analysis. USGS (The National Map Viewer) is a great source for DEMs. With a little help from research, picked three locations that may qualify as flat terrain (Everglades, Florida), hilly terrain (San Francisco, California), and mountainous terrain (Grand Canyon, Arizona). DEMs are 1/3 arc-second DEMs, 7.5 minute boxes, clipped to about a fourth of the size; already projected to NAD83. Created kernel text files using the paper, A Comparison of Two Approaches to Ranking Algorithms Used to Compute Hill Slopes, by Kevin H. Jones, as a reference. Created the text files using Python IDE, PyCharm.

- Everglades, Florida
    - Metadata:
        - Datum/CS: NAD 83
        - 5349 by 5295
        - Unit: meters
    - Topological map (for a sense of knowing; not analysis)

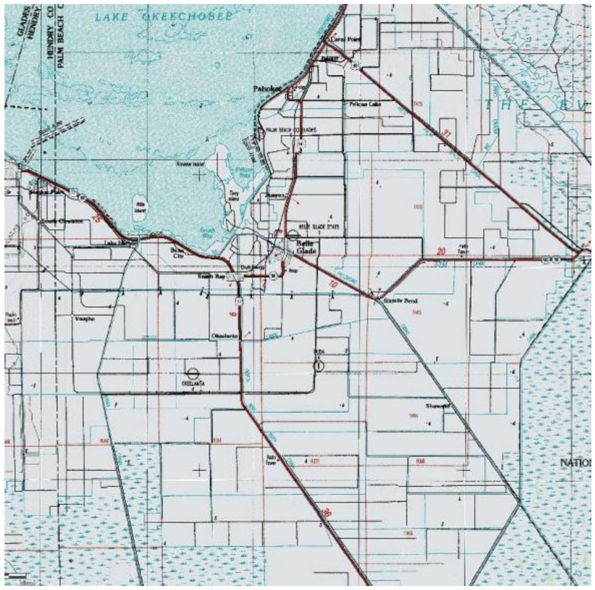


Figure 1: Topological map for Florida

- Where on the map is that?

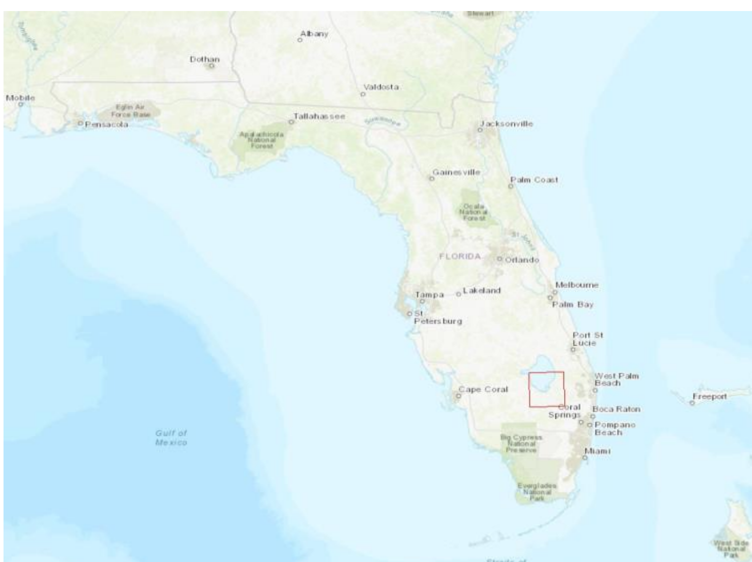


Figure 2: Screenshot of ArcMap basemap w/ AOI in red

- San Francisco, California
    - Metadata:
        - Datum/CS: NAD 83
        - 1130 by 1466
        - Unit: meters
    - Topological map (for a sense of knowing; not analysis)

Figure 3: Topological map of San Francisco

- Where on the map is that?

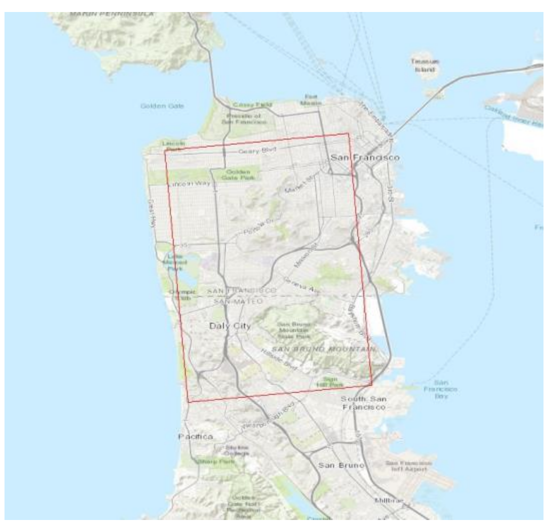


Figure 4: Screenshot of ArcMap basemap w/ AOI in red

- Grand Canyon, Arizona
    - Metadata:
        - Datum/CS: NAD 83
        - 5349 by 5295
        - Unit: meters
    - Topological map (for a sense of knowing; not analysis)

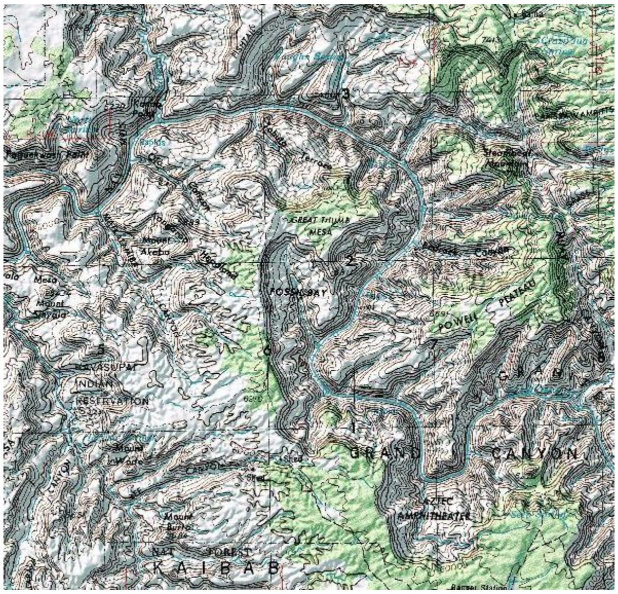


Figure 5: Topological map of Grand Canyon

- Where on the map is that?

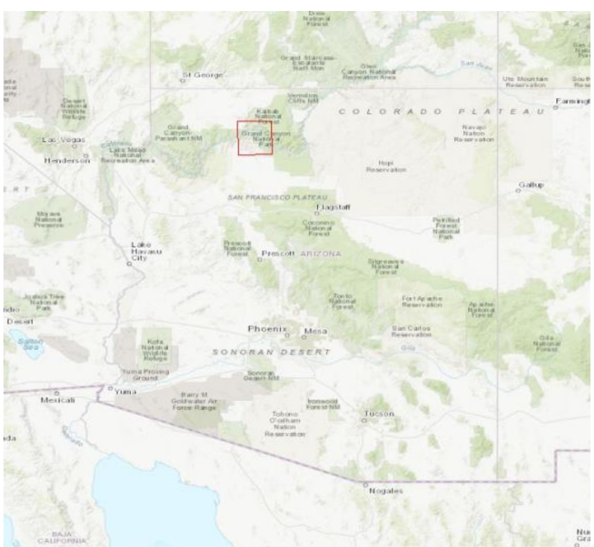


Figure 6: Screenshot of ArcMap basemap w/ AOI in red

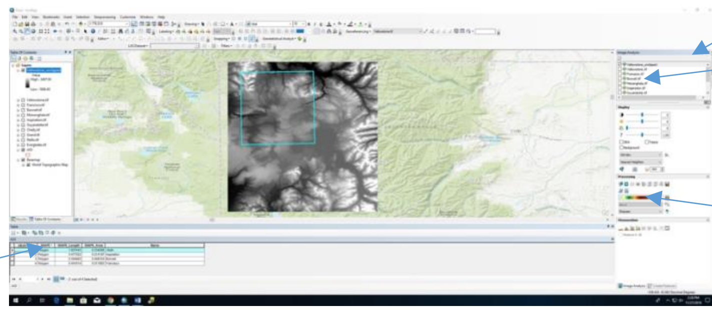
Step 2: Data Preprocessing

- Downloaded/Extracted files from USGS online
- Clip DEMs
    - Using image analysis tool
        - Select raster wanting to clip and polygon wanting to clip with (creates temp. file); digitized a polygon about a quarter of the original DEM
        - Save raster

Before:

Image analysis box

Select raster



Press clip button once polygon and raster are selected

Select polygon

Figure 7: Before clipping

After:

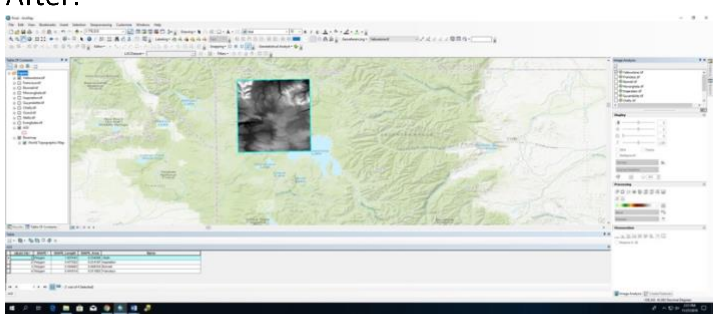


Figure 8: After clipping

- Slope Algorithms
    - Horn’s Method (Horn, 1981)
        - Nearest points weighted more than diagonal neighbors
        - Also known as the Sobel operator, which can be used for edge detection
        - The center cell actually has no influence on the calculated slope (maybe a disadvantage?)
    - Fleming & Hoffer (1979)
        - Probably next best algorithm, behind Horn’s
        - Takes dz/dx and dz/dy using cardinal directions
        - Center cell not taken into account for calculation

- Diagonal Ritters
  - Modification of Fleming and Hoffer's
  - Takes dz/dx and dz/dy using ordinal directions (hence the name)
  - Center cell not taken into account for calculation
- Simple Method
  - Uses the difference in elevation with its neighbor to calculate slope
  - Only method where center cell is actually taken into account
  - Ranks relatively low for slope calculations (wanted to add in for comparison anyways)

- Kernels
  - Horn's Method

```
3 3
0.125 0 -0.125
0.25 0 -0.25
0.125 0 -0.125
```

Figure 9: dz/dx (NS)

```
3 3
-0.125 -0.25 -0.125
0 0 0
0.125 0.25 0.125
```

Figure 10: dz/dy (EW)

  - Fleming & Hoffer

```
3 3
0 0 0
0.5 0 -0.5
0 0 0
```

Figure 11: dz/dx (NS)

```
3 3
0 -0.5 0
0 0 0
0 0.5 0
```

Figure 12: dz/dy (EW)

  - Diagonal Ritters

```
3 3
0.35355 0 0
0 0 0
0 0 -0.35355
```

Figure 13: dz/dx (NW-SE)

```
3 3
0 0 -0.35355
0 0 0
0.35355 0 0
```

Figure 14: dz/dy (NE-SW)

  - Simple Method

```
3 3
0 0 0
0 1 -1
0 0 0
```

Figure 15: dz/dx (NS)

```
3 3
0 -1 0
0 1 0
0 0 0
```

Figure 16: dz/dy (EW)

Step 3: ArcGIS Processing

ArcMap's focal statistics tool can be used to help find the slope of a DEM. Before going into the processing steps, here is some background information on the topic at hand.

What is a focal function?
As according to Dr. Helper's notes, focal functions are neighborhood functions that use values in adjacent cells to return values for a new raster, which can be used for aggregation, filtering, and/or computing slope and aspect.

What exactly is slope (as it pertains to ArcMap)?
Maximum rate of change in elevation between a cell and its eight neighbors identifies the steepest downhill slope. In this project, calculated slope using percent not degrees.

What is the moving window operation?
Also known as convolution, it can be used to apply filters to an image in order to manipulate its pixel values: slope, sharpening, softening, edge detection, etc.

What is a kernel and what do you do with it?
Matrix used to add weighted values to a single pixel and its neighbors when using convolution filters.

Four algorithms used: Horn's, Fleming & Hoffer's, Simple, and Diagonal Ritters.
As discussed above in step 2.

How do you even calculate slope?
On the ESRI website where documentation can be located, it shows calculations for slope, both in degrees and percent. For this project, slope will be calculated in percent.

Here are visuals for slope and kernel calculations.

A	B	C
D	E	F
G	H	I

Figure 17: Kernel representation

$$\frac{dz}{dx} = \frac{(c + 2f + i) - (a + 2d + g)}{8}$$

$$\frac{dz}{dy} = \frac{(g + 2h + i) - (a + 2b + c)}{8}$$

Figure 18: formula translates to the proper kernel for Horn's method

$$\frac{dz}{dx} = \frac{f-d}{2}$$

$$\frac{dz}{dy} = \frac{b-h}{2}$$

Figure 19: Fleming & Hoffer's

$$slope = \sqrt{\frac{dz}{dx}^2 + \frac{dz}{dy}^2}$$

Figure 20: calculating slope using rate of change data (dz//dx, dz/dy)

With three different DEMs and four different algorithms, running all the necessary tools can get really confusing, really fast. There were a couple of solutions to this problem; either: a) write a python script, or b) use model builder. In the case of python, not all tools transfer over to arcpy smoothly. One in particular…raster calculator. Raster calculator does not exist in arcpy. So in this case, for this particular project, the best decision was to go with model builder.

AOI: ~a quarter of the original DEM. Digitized a box and used to clip for all locations.

Model Builder allows creation of a new tool, using the ArcGIS toolbox.
Here is an example of the processing tools used for each DEM and algorithm:

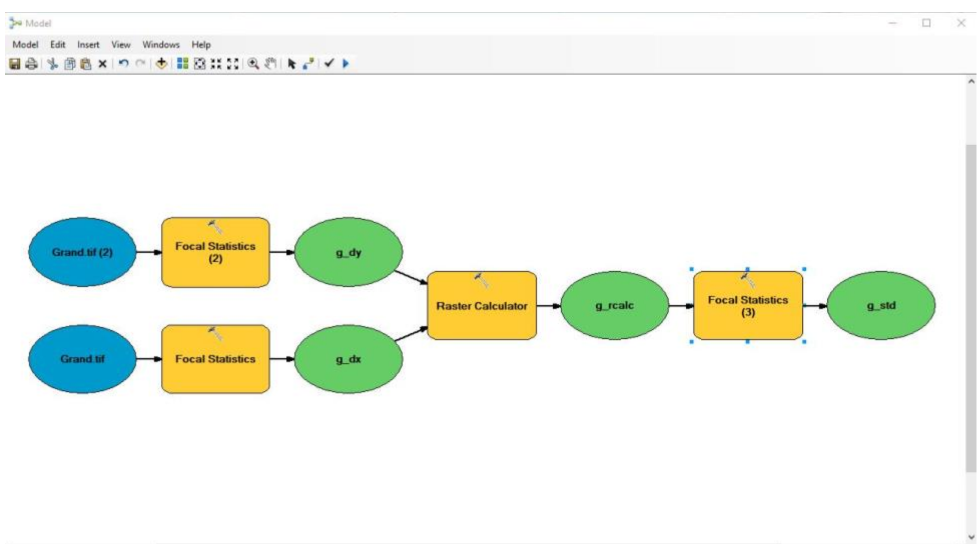


Figure 21: Model Builder

This tool/model was ran 12 times, one time per DEM per algorithm.

In this model, approximately four geoprocessing tools are ran. This is the case for one DEM per algorithm; 3 DEMs times 4 algorithms times 4 geoprocessing tools, comes out to be 48 tools ran, not including the preprocessing steps. Python would've been very beneficial in this situation, except having to find a work around situation for raster calculator. Using model builder though did help relieve some of the work, by going from 48 tools ran, to just 12.

- Focal Statistics
    - This tool is able to calculate the partial direction for both north-south and east-west. These numbers are needed in order to calculate the slope.
    - In order to get these numbers, one must first create the right kernel for an algorithm; in doing so, refer to this [paper] (also mentioned in the introduction)
    - Once the tool is open, parameters need to be set to as the following:

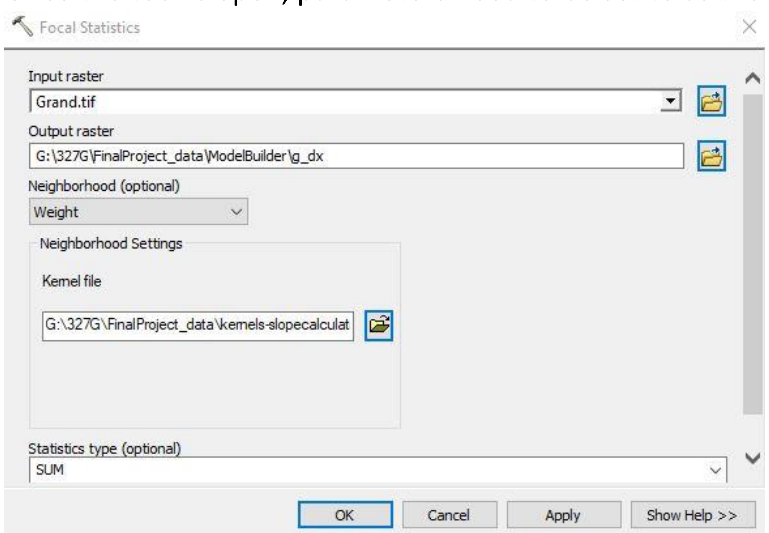


Figure 22: focal statistics calculation for dz/dx and dz/dy

    - Kernel file being the particular one for the correct direction and algorithm
- Raster Calculator
    - Once both x rate of change, and y rate of change have been calculated, then use the raster calculator to calculate slope using the formula above

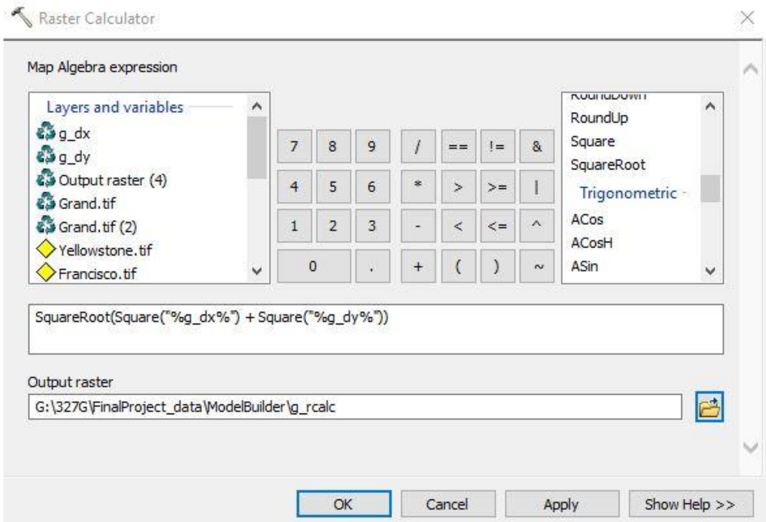


Figure 23: raster calculation

- Focal Statistics
    - Going one step further, calculate the standard deviation using focal statistics
    - This will allow more analysis over results

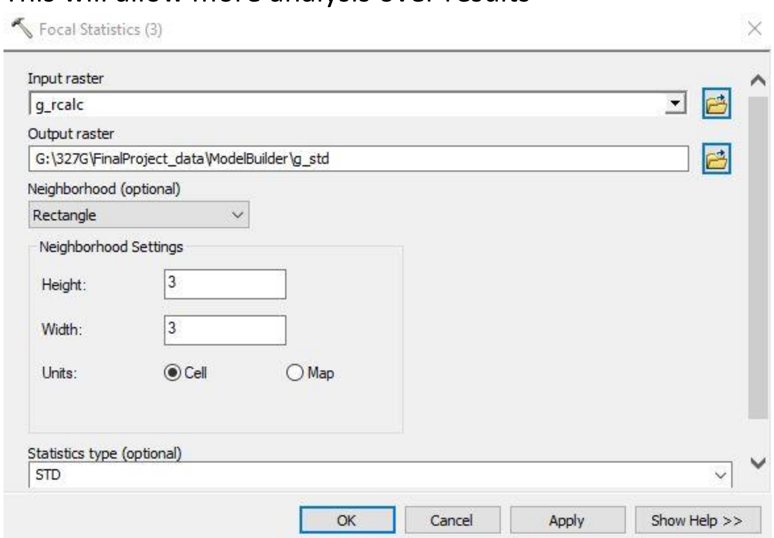


Figure 24: STD calculation

Step 4: Data Presentation

Below are the outputs from all of the DEMs created.

Fleming & Hoffer	Elevation (m)	Slope (max)	STD (relative value)
Everglades	28.1161 – (-2.20066)	9.87327	3.31226
San Francisco	399.963 – 2.29147	159.937	69.4942
Grand Canyon	2461.65 – 547.224	1740.24	763.704

Table 1

Horn Method	Elevation (m)	Slope (max)	STD (relative value)
Everglades	28.1161 – (-2.20066)	8.63692	3.19169
San Francisco	399.963 – 2.29147	126.518	54.2024
Grand Canyon	2461.65 – 547.224	1375.69	596.265

Table 2

Simple Method	Elevation (m)	Slope (max)	STD (relative value)
Everglades	28.1161 – (-2.20066)	13.3609	5.02835
San Francisco	399.963 – 2.29147	318.95	130.485
Grand Canyon	2461.65 – 547.224	3309.39	1440.05

Table 3

Diagonal Ritters	Elevation (m)	Slope (max)	STD (relative value)
Everglades	28.1161 – (-2.20066)	8.43528	3.09966
San Francisco	399.963 – 2.29147	113.177	51.7052
Grand Canyon	2461.65 – 547.224	1230.76	583.065

Table 4

Slope tool	Elevation (m)	Slope (max)	STD (relative value)
Everglades	28.1161 – (-2.20066)	84.0349	31.0543
San Francisco	399.963 – 2.29147	160.877	51.0929
Grand Canyon	2461.65 – 547.224	1422.6	545.238

Table 5

Step 5: Analysis

With the knowledge that Horn's Method is the default in ArcMap, one can only assume that this would be the best slope algorithm for the choosing. After running the geoprocessing tools, we will be able to determine how true this actually might be.

What do the numbers on the tables from step 4 mean?
Gray box: This the case for the first four boxes, it is the type of algorithm used for those set of numbers. The gray box on the last table says slope tool; these are the results from running the slope geoprocessing tool on the clipped rasters.
1st column: Names of the locations where the rasters are from.
2nd column: Elevation (m) values from the original clipped raster; same numbers for all five tables
3rd column: Max value of the slope/'raster calculation' raster; The new DEM, after running the slope calculation, now has slope values that fill its cells (instead of elevation).
4th column: Max standard deviation value calculated using the focal statistics. These numbers will allow visible results, being that using max slope wasn't very applicable since the max slope could be a huge outliner.

Analysis will involve comparing standard deviation rasters.
Standard deviation is a number that measures how far spread out data is from the mean. A low s.d. means that most of the numbers are very close to the average. A high s.d. means that the numbers are spread out.

Based on visuals and tables:

Slope algorithms ranked for flat terrain, best to worst:
-Fleming & Hoffer's

-Horn Method
-Diagonal Ritters
-Simple Method

Slope algorithms ranked for hilly terrain, best to worst:
-Horn Method
-Fleming & Hoffer's
-Diagonal Ritters
-Simple Method

Slope algorithms ranked for mountainous terrain, best to worst:
-Horn Method
-Fleming & Hoffer's
-Diagonal Ritters
-Simple Method

Analysis also included in the Conclusion section.

Conclusion

Per type of terrain, which slope algorithm would be the best to calculate terrain with? And what are the effects of the other methods?

Different algorithms can be more appropriate for different purposes because each algorithm prioritizes different values in their calculations.
(Conclusions based on slope-calculated rasters (subjective) and tables (objective, measurable); rasters not pictured.)

Averaging techniques such as the Horn algorithm, since it doesn't use the center cell in calculations, can lose accuracy at that immediate cell. Features such as peaks, ridges, depressions, and valleys have the potential to be smoothed over. Algorithms that give weight to maximum slope tend to overestimate slope, and depending on which of the 8 neighbor cells are used in the calculation, slope will be more accurate in that direction.

Looking at low terrain variability, Fleming & Hoffer's algorithm produced the clearest looking product, suggesting that it would be suited for smoother, flatter terrain. Fleming and Hoffer's seems to be more effective for smooth surfaces and can serve as an alternative to the Horn algorithm in areas that have a smaller slope differences. Despite concerns about accuracy on the center cell, the Horn algorithm placed the most emphasis in valleys, pits, ridges, and peaks than other methods. Horn algorithm seems to be especially effective for areas characterized by high variations in slope.

Diagonal Ritter's has similar properties to Fleming & Hoffer's. Uses ordinal neighbors instead of cardinal (nearest) neighbors.

Simple method did not offer any type of credible results. Following by its name, it's a simple solution to calculating slope, but not a very accurate one.

Limitations & Improvements

- Resolution of Imagery - effectiveness of each algorithm might be changed by improvements or reductions to cell resolution
- Oversimplification of classifications of "low", "medium", "high" variations in slope - What is defined as flat/hilly/mountainous terrain?
- Difficult to determine a one-size fits all algorithm to be definitely "better" than others.
- RMS error could be calculated to compare algorithms for accuracy