Environment and organisms - Task #411

Insert DEM adaptive smoothing procedure into terrain workflow

05/15/2012 09:52 AM - Jim Regetz

Status:	In Progress	**Start date:**	03/15/2012
Priority:	Normal	**Due date:**	
Assignee:	Jim Regetz	**% Done:**	30%
Category:	Terrain	**Estimated time:**	40.00 hours
Target version:			
Activity type:			

Description

The original Oregon terrain processing workflow, implemented as a set of AML scripts, included the application of John Gallant's adaptive smoothing procedure to the input SRTM layer. We need to decide how best to integrate some form of this procedure into our current terrain layer development workflow. Some specific sub-tasks:

- *Decide which DEMs to use as input.* Per past comments from John, it should probably be applied separately to SRTM and ASTER GDEM1 because they have different noise characteristics, but we still need to decide whether to do it before or after resampling them to our 90m grid.
- *Decide how to characterize noise.* The procedure requires specification of either a constant noise standard deviation or a full grid specifying pixel-level SD. In either case, we need to come up with this for both SRTM and GDEM2.
- *Liberate from ArcINFO.* We should be able to do this smoothing without the dual constraints of having to run on Windows and deal with ESRI licensing.

History

#1 - 05/15/2012 11:22 AM - Jim Regetz

- Status changed from New to In Progress

- Start date changed from 05/15/2012 to 03/15/2012

- % Done changed from 0 to 30

I developed two alternative implementations of the adaptive DEM smoothing procedure (multiscalesmooth9a_clean.aml):

- GRASS Python module (source:terrain/research/oregon/multiscalesmooth.py)
- R script, leveraging the raster package (source:terrain/research/oregon/multiscalesmooth.R)

For reasons of performance and scalability, I would regard the GRASS version as authoritative over the R version. Using the 144M pixel Oregon SRTM case study input with constant noise standard deviation of 0.001, the GRASS procedure completes in 10-20 minutes (depending on compression settings), which is comparable to the original AML routine. The R script is tolerable when the input raster is small enough that all processing can be done in memory, but is far to slow when processing is done via disk.

Outputs of my code vs the original AML using the Oregon SRTM are qualitatively the same but not perfectly identical, I believe for the following reasons:

1. Depending on the dimensions of the input raster, the procedure needs to expand the raster extents in anticipation of the multilevel aggregation. My code does this in a slightly different way, meaning that the aggregation neighborhoods in my implementations may not be exactly the same as in the AML implementation. This can yield minor differences in the smoothed output, especially immediately along edges and coasts.
2. The AML implementation uses single-precision floating point values for all intermediate (and final) grids, a limitation imposed by Arc/INFO GRID format. GRASS and R can both use double precision. This can make a numerical difference, although from what I've seen it's mostly (entirely?) inconsequential if the final output DEM is rounded back to integer (meters).