Received: 30 June 2022 | Revised: 13 April 2023 | Accepted: 18 April 2023

DOI: 10.1111/tgis.13067

Transactions in GIS WILEY

RESEARCH ARTICLE

MultiscaleDTM: An open-source R package for multiscale geomorphometric analysis

Alexander R. Ilich[1] | Benjamin Misiuk[2] | Vincent Lecours[3,4] | Steven A. Murawski[1]

[1]College of Marine Science, University of South Florida, St. Petersburg, Florida, USA

[2]Department of Oceanography, Dalhousie University, Halifax, Nova Scotia, Canada

[3]Geography Program, Université du Québec à Chicoutimi, Chicoutimi, Quebec, Canada

[4]School of Forest, Fisheries, and Geomatics Sciences, University of Florida, Gainesville, Florida, USA

Correspondence
Alexander R. Ilich, College of Marine Science, University of South Florida, St. Petersburg, FL, USA.
Email: ailich@usf.edu

Funding information
National Fish and Wildlife Foundation, Grant/Award Number: GEBF45892; National Oceanic and Atmospheric Administration, Grant/Award Number: NA20NOS4000227

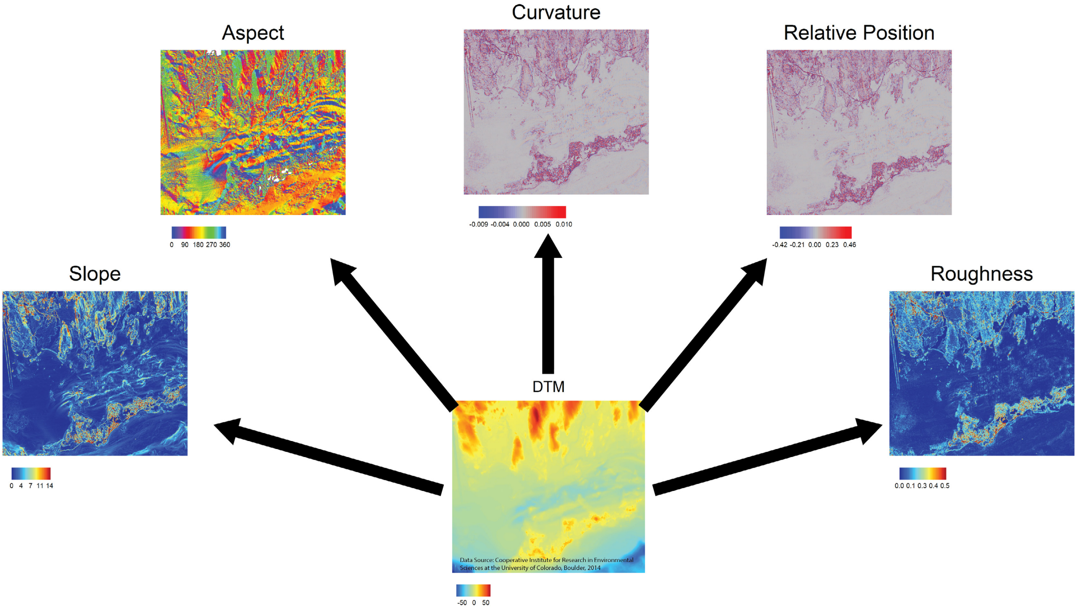
Abstract

Digital terrain models (DTMs) are datasets containing altitude values above or below a reference level, such as a reference ellipsoid or a tidal datum over geographic space, often in the form of a regularly gridded raster. They can be used to calculate terrain attributes that describe the shape and characteristics of topographic surfaces. Calculating these terrain attributes often requires multiple software packages that can be expensive and specialized. We have created a free, open-source R package, `MultiscaleDTM`, that allows for the calculation of members from each of the five major thematic groups of terrain attributes: slope, aspect, curvature, relative position, and roughness, from a regularly gridded DTM. Furthermore, these attributes can be calculated at multiple spatial scales of analysis, a key feature that is missing from many other packages. Here, we demonstrate the functionality of the package and provide a simulation exploring the relationship between slope and roughness. When roughness measures do not account for slope, these attributes exhibit a strong positive correlation. To minimize this correlation, we propose a new roughness measure called adjusted standard deviation. In most scenarios tested, this measure produced the lowest rank correlation with slope out of all the roughness measures tested. Lastly, the simulation shows that some existing roughness measures from the literature that are supposed to be independent of slope can actually exhibit a strong inverse relationship with the slope in some cases.

1 | INTRODUCTION

Digital terrain models (DTMs) are datasets containing altitude values above or below a reference level, such as a reference ellipsoid or a tidal datum over geographic space, often in the form of a regularly gridded raster, which comprise values stored within regularly gridded “cells” or “pixels” (Katzil & Doytsher, 2000; Krcho, 1999; Lecours et al., 2016). DTMs form the backbone of quantitative geomorphometry and are used in a wide variety of applications across various scientific disciplines, some of which include environmental and earth sciences, engineering, and oceanography (Pike et al., 2009). Terrain attributes that describe the shape and character of the Earth's surface can be calculated from these DTMs. While a large number of attributes can be found in the literature (Wilson, 2018), the most common terrain attributes can generally be categorized into five thematic groups: slope, aspect, curvature, relative position, and roughness (Bouchet et al., 2015; Lecours et al., 2016, 2017; Wilson et al., 2007) (Figure 1). While this framework is by no means the only way to group terrain attributes, it provides a useful way to organize related terrain attributes based on their physical meaning and will be used within this article for organizational purposes.

Although widely used, the selection and application of terrain attributes to a particular task may often be suboptimal (Ironside et al., 2018; Lecours et al., 2017). Owing to the abundance of attributes found in the literature and measures being dispersed across various software packages, users may not fully understand how the terrain attributes they select are calculated, and may only choose a subset of readily available attributes, which can result in a failure to fully capture useful information that could be derived from the DTM (Bouchet et al., 2015; Lecours et al., 2017). For example, fields such as ecology often use terrain attributes as predictors for species distribution models (Franklin, 1995; Ironside et al., 2018; Mod et al., 2016). However, many scientists are unaware of the



Data Source: Cooperative Institute for Research in Environmental Sciences at the University of Colorado, Boulder, 2014

FIGURE 1 The five thematic groups of terrain attributes that can be derived from a digital terrain model are slope, aspect, curvature, relative position, and roughness. Figure adapted from Wilson et al. (2007).

full suite of metrics available to them (Bouchet et al., 2015), some of which may only be available in specialized software (Wilson, 2018). These issues are compounded by confusion in terminology within the literature, and at times, opaque documentation—particularly where variable results are obtained using different software packages (Bishop et al., 2012; Bouchet et al., 2015; Lecours et al., 2017; Minár et al., 2020; Sofia, 2020).

Additional complexity arises from scale dependence, wherein terrain attributes vary based on the scale at which they are calculated (Dolan, 2012; Misiuk et al., 2021; Moudrý et al., 2019; Wilson et al., 2007; Wood, 1996), and many software packages allow for calculation only at a fixed scale (Lecours et al., 2017; Misiuk et al., 2021). Smoothing filters (e.g., focal mean) are commonly applied to either the DTM or terrain attributes to achieve a broader analysis scale, which is possible with almost any spatial data software (Dolan, 2012; Misiuk et al., 2021; Moudrý et al., 2019). However, many terrain attributes can also be directly calculated by varying the size of the focal window (Figure 2), which may be preferable in many cases (Misiuk et al., 2021).

Focal analysis (also known as neighborhood analysis) works by moving one cell at a time through a raster image, extracting values from nearby cells within a specified $n \times n$ cell window (Figure 2), performing some operation on those values, and then assigning the result of this operation to the corresponding position of the window's central cell in a new raster image of the same dimensions. Many software packages will restrict focal calculations to a 3×3 cell window; however, in principle n can be any odd integer. Often n is set to be equal in both the x and y direction resulting in a square window, but focal windows can also be rectangular. Moreover, values in the focal window can be excluded from the calculation to provide a greater variety of shapes such as a circle or annulus (Figure 3), or in some cases, only a subset of the data within the window may be required for calculation (Figure 4).

While window size is useful for varying the spatial scale of analysis, it is important to recognize that terrain attributes calculated on different surfaces with the same window size are not necessarily comparable across datasets if the window size is measured in cell units (e.g., 3×3 cell focal window). For example, a 3×3 cell slope calculation on a 1 m resolution DTM and a 3×3 cell slope calculation on a 10 m resolution DTM represent different spatial scales. When comparing measures conducted on different surfaces and with different methods it is important to

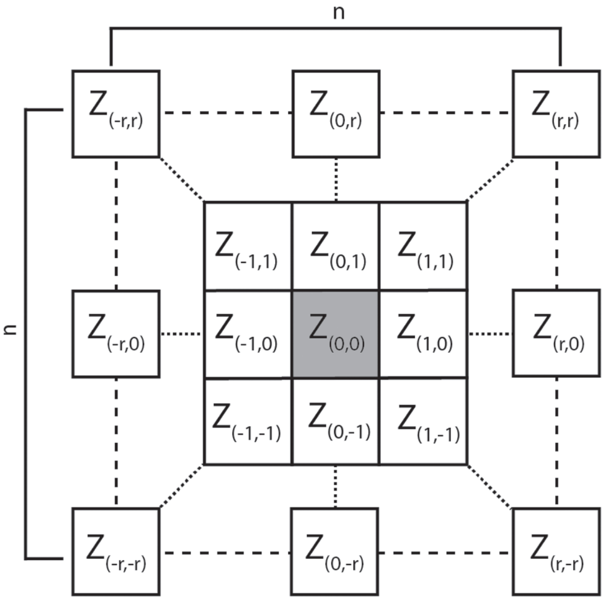


FIGURE 2 Schematic for a variable-size focal window. The version shown here is an $n \times n$ square window which is the most common; however, focal windows can also be rectangular having unequal n in the x and y directions. n represents the number of cells in a given dimension, and r represents the half-width or radius of the focal window, therefore, $n = 2r + 1$. Figure adapted from Wilson et al. (2007).

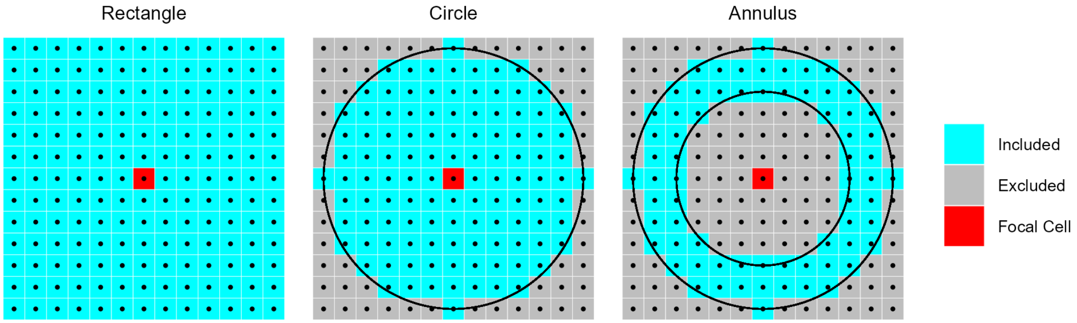


FIGURE 3 Examples of different focal window shapes. Shown are a 13 × 13 cell rectangular window (left), a circular window with a radius of six cells (center), and an annulus window with an inner radius of four cells and an outer radius of six cells (right).

use transferable parameters such as the "total analysis distance," which is "the maximum *x* and *y* distance in map units over which the value of the terrain attribute at the focal cell is affected" (Misiuk et al., 2021) and is in many cases simply the size of the focal window expressed in map units.

Here, we present an R package called `MultiscaleDTM` (Ilich et al., 2021), which enables the calculation of several terrain attributes within each of the five thematic groups (slope, aspect, curvature, relative position, and roughness) using a variable window size (Table 1). The variable window size method is currently not available for all types of terrain attributes in many software packages, which has limited its widespread adoption (Lecours et al., 2017; Misiuk et al., 2021). R (R Core Team, 2022) is a free and open-source programming language that is widely used by scientists across various disciplines (Tippmann, 2015), including geomorphometry (Grohmann, 2004; Hengl, 2011) and ecology (Lai et al., 2019), that has developed substantial support for spatial data analysis over the last decade and a half (Bivand, 2021; Lovelace et al., 2019). Despite support for spatial data, there is currently no other comprehensive R package for calculating measures from all of the five thematic groups of terrain attributes at multiple different spatial scales of analysis. Implementation of this functionality in a free, transparent, flexible, comprehensive, and well-documented tool is motivated by a desire to increase accessibility to scientists from different backgrounds and in various disciplines and enable more objective and reproducible results.

In this article, we will first describe the different terrain attributes that can be calculated by `MultiscaleDTM` as well as their corresponding functions. These will be organized into five thematic groups: slope, aspect, curvature, roughness, and relative position. Slope, aspect, and curvature will be described in a single section since all of these measures rely on surface derivatives and, therefore, can be calculated simultaneously. After describing the available measures, we will briefly describe some details regarding the design and implementation of the software. Next, we will provide a demonstration of several functions on a real-world data set. Then lastly we will examine the performance of various roughness measures via simulation to assess their relationship to slope.

2 | METHODS

2.1 | Terrain attribute functions

2.1.1 | Slope, aspect, and curvature

Slope and aspect can be calculated from first-order partial derivatives of a topographic surface (Olaya, 2009). The slope is a measure of the steepness of the focal cell and is obtained by calculating the angle between

TABLE 1 List of available terrain attributes in `MultiscaleDTM` and their definitions.

Terrain attribute	Functions	Thematic group	Description
Slope	SlpAsp, Qfit	Slope	The maximum rate of change in elevation values
Aspect	SlpAsp, Qfit	Aspect	The orientation of the slope in the downslope direction
Eastness	SlpAsp, Qfit	Aspect	The east/west components of the orientation of the slope are calculated as the sine of aspect. Thus, it ranges between −1 (due West) and 1 (due East)
Northness	SlpAsp, Qfit	Aspect	The north/south components of the orientation of the slope are calculated as the cosine of aspect. Thus, it ranges between −1 (due South) and 1 (due North)
Profile curvature	Qfit	Curvature	The level of convexity or concavity along the direction of the maximum slope
Planform curvature	Qfit	Curvature	The level of convexity or concavity perpendicular to the direction of the maximum slope
Twisting curvature	Qfit	Curvature	The amount of local twisting is measured as the change in slope angle per unit distance along the direction perpendicular to the slope
Minimum curvature	Qfit	Curvature	The minimum curvature in any plane
Maximum curvature	Qfit	Curvature	The maximum curvature in any plane
Mean curvature	Qfit	Curvature	The average of the minimum and maximum curvatures
Relative position	RelPos, DMV, TPI, BPI	Relative position	An indication of whether an area is a local topographic high (positive values) or low (negative values). It can be measured in either units of the DTM or standardized based on local topography to be a unitless measure
Vector ruggedness measure	VRM	Roughness	The dispersion of unit vectors normal to the terrain surface
Surface area to planar area ratio	SAPA	Roughness	The surface area divided by the planar area
Adjusted standard deviation	AdjSD	Roughness	The local standard deviation of elevation values in a focal window after removing the influence of slope by fitting a plane to the focal window using ordinary least squares and extracting the residuals
Roughness index-elevation	RIE	Roughness	The local standard deviation of the residual topography surface, where the residual topography surface is calculated as the DTM minus the focal mean of the DTM

the horizontal plane and the one tangential to the surface, while aspect indicates the downslope direction (Olaya, 2009). Slope and aspect are typically measured in degrees or radians, but slope may also be reported as a percent. For aspect, we adopt the convention that 0° represents a due North direction and increases clockwise such that due East is at 90°, due South at 180°, and due West is at 270°. It is often improper or difficult to use aspect directly in statistical models since degrees are on a circular scale, meaning that distant values can actually be similar in what they represent. For example, aspect values of 359° and 1° have a difference of 358° when in fact they represent a difference in direction of only 2°. To ensure a valid integration into statistical analyses, aspect is, therefore, often decomposed into the eastness and northness. Eastness is calculated as the sin(aspect) and indicates the east/west direction of slope, and the northness is calculated as the cos(aspect) and indicates the north/south direction of the slope (Wilson et al., 2007). Eastness and northness range from −1 to 1, where −1 represents downslope directions of due West and due South, and 1 represents downslope directions of due East and due North, respectively.

In addition to slope and aspect, curvature can be calculated from a DTM based on the second-order partial derivatives (Olaya, 2009). There are a large number of curvatures that can be defined and Minár et al. (2020) highlight a lack of consistency in the scientific literature and across software packages regarding the meaning, formulas, sign convention, and units for various curvature measures. These curvatures however can be categorized into three basic groups: profile (along slope), planform (across slope), and twisting curvatures. For curvature measures, we adopt a geographic sign convention where convex is positive and concave is negative (i.e., hills are considered convex with positive curvature values) (Minár et al., 2020).

SlpAsp

The `SlpAsp` function calculates multiscale slope and aspect according to the method proposed by Misiuk et al. (2021) (Appendix A), which is a direct modification of the traditional 3 × 3 slope and aspect algorithms for the "rook" (4 neighbors) (Fleming & Hoffer, 1979; Ritter, 1987) and "queen" (8 neighbor case) (Horn, 1981). This algorithm, therefore, only considers a subset of cells within the focal window—specifically, the four cells on the edge of the focal window directly up, down, left, and right of the focal cell for the "rook" case and an additional four corner cells for the "queen" case (Figure 4). It is worth

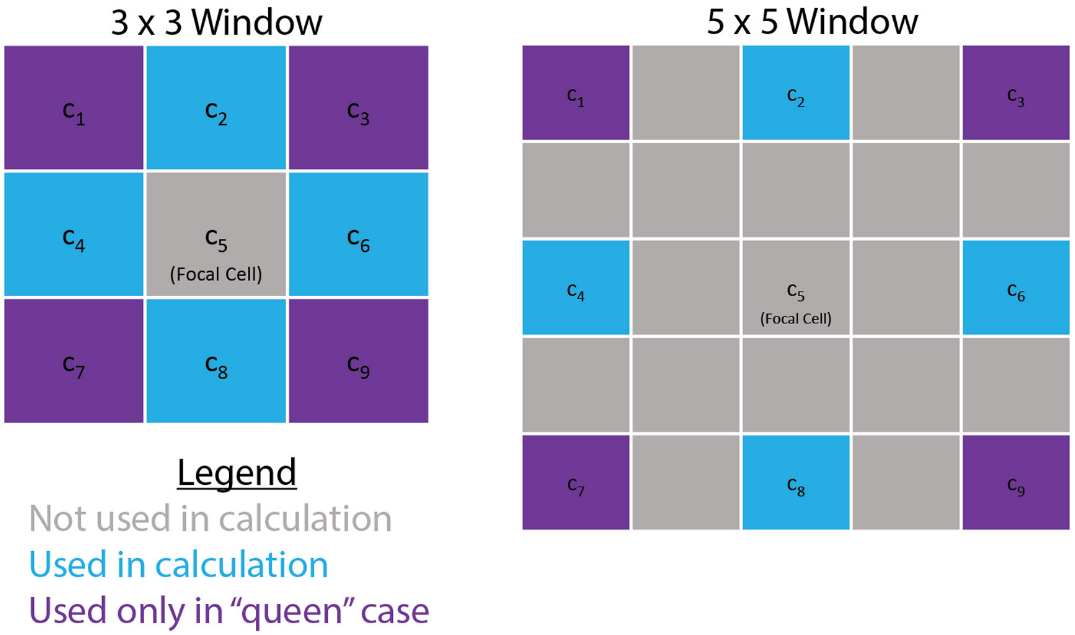


FIGURE 4 Schematic for cells used in the Misiuk et al. (2021) multiscale slope and aspect algorithm.

noting that since only edge cells are used in this method, successively larger windows do not contain data from smaller window sizes so that different scales are not inherently correlated with one another and it is computationally less demanding than alternative methods that use all values in the window (e.g., see `Qfit` below).

Qfit

The `Qfit` function fits a quadratic surface to a rectangular focal window according to Equation (1) using ordinary least squares (Figure 5) (Evans, 1980; Wood, 1996), where *a*–*f* are regression parameters, *Z* is the elevation/depth, *X* is the east/west coordinates in the focal window relative to the focal cell, and *Y* is the north/south coordinates in the focal window relative to the focal cell. The fitted regression parameters are then used to calculate slope, aspect, and various types of curvature (Appendix A) (Evans, 1980; Wood, 1996). As mentioned above, the three basic groups of curvatures are profile (along slope), planform (across slope), and twisting (Minár et al., 2020). The `Qfit` function calculates the "principal representatives" of each curvature group, which are normal slope line curvature ($(k_n)_s$) (Evans, 1980; Minár et al., 2020), normal contour curvature ($(k_n)_c$) (Krcho, 1983; Minár et al., 2020), and contour geodesic torsion ($\tau(g)_c$) (Minár et al., 2020), respectively. Additionally, the function calculates the maximum ($k_{\max}$), minimum ($k_{\min}$), and mean curvatures (k_{mean}) (Minár et al., 2020; Shary, 1995). Note that all curvatures are measured in units of length^{-1} (e.g., m^{-1}) except twisting curvature, which is measured in rad length^{-1} (i.e., change in angle in radians per unit distance). `Qfit` provides a more general solution than `SlpAsp` as it evaluates the second derivatives of the local surface and can, therefore, calculate curvatues; however, this along with the use of all of the values in the focal window rather than a subset of the values makes it more computationally expensive.

In addition to calculating slope, aspect, and measures of curvature, `Qfit` can define seven discrete landform classes known as morphometric features based on some of these measures and a user-specified tolerance threshold for slope and curvature that define a flat/planar surface (Wood, 1996). These classes are planar flat, planar slope, pit, channel, pass, ridge, and peak (Appendix B).

Additional features of the function include options to calculate a smoothed version of the topographic surface (i.e., the intercept term *f*), return the main regression parameters (*a*–*e*), or return the various partial derivatives so that other terrain attributes can be calculated. If partial derivatives are returned they can be used directly in

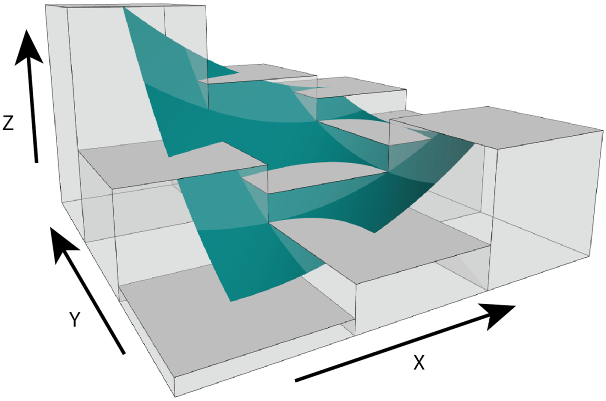


FIGURE 5 Quadratic surface representation of digital terrain model data within a focal window. Figure adapted from Walbridge et al. (2018).

the equations provided by Minár et al. (2020) for additional measures of curvature. Finally, the surface can be constrained to pass exactly through the focal cell by removing the intercept term, *f*, in Equation (1) and subtracting the elevation/depth value of the focal cell from all values in the focal window, such that the focal cell's *XYZ* coordinates are defined by (0, 0, 0) (Wood, 1996).

An interactive 3D R Shiny application can also be launched using the function `explore_terrain()` to aid in the understanding of the different measures that can be calculated by `Qfit`. In this application, the regression parameters of the quadratic surface can be changed and the resultant 3D surface will be plotted and the values of each terrain attribute are displayed for that local window. The application is also available online at https://ailich.shinyapps.io/Terrain_Attributes_Explorer_App/.

$$Z = aX^2 + bY^2 + cXY + dX + eY + f \quad (1)$$

2.1.2 | Relative position

Relative position indicates whether an area is a local topographic high (positive values) or low (negative values) in relation to a reference height. This is typically accomplished by calculating the difference between the value of a focal cell and the mean of included values in the focal window, but other functions such as the median, minimum, or maximum of included cells may be appropriate depending on the application. Relative position can be expressed in units of the input DTM raster or can be standardized relative to the local topography by dividing by the standard deviation or range of included elevation values in the focal window (Lecours et al., 2017; Wilson & Gallant, 2000). Measures of the relative position include the difference from mean value (DMV) (Lecours et al., 2017; Wilson & Gallant, 2000), topographic position index (TPI) (Weiss, 2001), and bathymetric position index (BPI) (Lundblad et al., 2006). DMV, TPI, and BPI all use the mean of included values in the focal window as the reference height. These measures are strongly related and mainly differ based on the shape of the focal window used. Additionally, there is some ambiguity in their meaning. For example, TPI and DMV have been found to be equivalent when calculated in SAGA GIS (Lecours et al., 2017), but these values differ from TPI as calculated from GDAL. This is because for TPI, SAGA GIS includes the central cell in the calculation of the mean, whereas in GDAL the central cell is excluded (i.e., GDAL defines TPI as the elevation of the focal cell relative to the mean of its surroundings). In `MultiscaleDTM`, we differentiate between DMV and TPI in that DMV includes the central cell and the TPI excludes the central cell in the focal calculations (i.e., calculation of local mean, standard deviation, and range). These measures can be calculated with the `TPI` and `DMV` functions, respectively. Both the `DMV` and `TPI` functions can use rectangular or circular focal windows (Figure 3). When an annulus-shaped focal window is used, this measure is often referred to as the bathymetric position index (BPI) (Lundblad et al., 2006). Although BPI is specified as "bathymetric," this is due to the context in which it was proposed, and it is equally applicable to terrestrial elevation data. BPI can be calculated with the `BPI` function. Since an annulus-shaped window is used, the `BPI` function requires an inner and outer radius to be specified such that cells are considered within the annulus only if the distance from the focal cell is greater than or equal to the inner radius and less than or equal to the outer radius (Figure 3). The annulus-shaped window allows for excluding cells near the focal cell and can, therefore, be designed in such a way that successively broader spatial scales do not contain any information from the finer scales and are, therefore, not inherently correlated.

In addition to the three relative position functions mentioned above, the `RelPos` function acts as a flexible and general-purpose function for calculating relative position. It allows for rectangular, circular, annulus, and even custom-shaped focal windows, as well as the ability to include or exclude the central cell from focal calculations (Figure 3). `RelPos` also allows for the specification whether the reference value

TABLE 2 Table of what capabilities are supported by each function at time of publication.

Function	Multiscale	Rectangular window	Circle window	Annulus window	Custom window	Unprojected coordinate system
SlpAsp	✓	✓				
Qfit	✓	✓				
DMV	✓	✓	✓			✓
TPI	✓	✓	✓			✓
BPI	✓			✓		✓
RelPos	✓	✓	✓	✓	✓	✓
VRM	✓	✓				✓
SAPA	✓	✓				
AdjSD	✓	✓				
RIE	✓	✓				✓

Note: "✓" indicates that it is supported by the function.

should be the mean, median, minimum, or maximum of included values in the focal window. `DMV`, `TPI`, and `BPI` are all calls to `RelPos` with different default parameter values. All relative position functions work for both projected or unprojected data (Table 2) and include the ability to standardize values based on local topography by dividing by the standard deviation or range of included elevation values in the focal window (Lecours et al., 2017; Wilson & Gallant, 2000). Some other software implementations standardize relative position by first calculating the raw (i.e., unstandardized) relative position values and then calculating a single "global" mean and standard deviation from the resultant surface. This global mean is then subtracted from the raw relative position values and then this is divided by the global standard deviation in an attempt to make the measure less sensitive to changes in spatial scale since the range of values tends to increase with increasing window size (Lundblad et al., 2006; Walbridge et al., 2018; Weiss, 2001); however, it is stated that this method should only be used if the original relative position surface has a mean "reasonably" close to zero (Weiss, 2001). If this global mean is substantially different from zero this could change the sign of many values and then remove the ability to interpret positive values as peaks and negative values as depressions. Therefore, this method of standardizing surfaces is not implemented directly in `MultiscaleDTM`, and instead, we implement standardization based on dividing by the focal standard deviation or range of elevation/depth values.

2.1.3 | Roughness

The term surface roughness (herein referred to as "roughness") is often used interchangeably with several other terms including ruggedness, rugosity, microrelief, and microtopography (Grohmann et al., 2011). While intuitively simple, there is no single precise definition of roughness, and, therefore, it can be conceptualized in many different ways (Grohmann et al., 2011; Hobson, 1972; Smith, 2014; Wilson et al., 2007). Despite the lack of a precise definition, roughness is some measure of the topographic variability at a given scale (Grohmann et al., 2011). It is important though that roughness measures differentiate between variability related to the slope and variability related to the "bumpiness" of the surface. However, many measures from the literature do not distinguish between the two sources, and as such a smooth but inclined surface will be considered rough, resulting in roughness measures that are inherently confounded with slope (Du Preez, 2015; Grohmann et al., 2011; Ironside et al., 2018; Sappington et al., 2007). The conflation of roughness with slope may lead to a violation of model assumptions (e.g., multicollinearity) or spurious statistical inference (e.g., regarding variable importance or interpretation of parameters). `MultiscaleDTM` only includes measures of roughness that account for the influence of slope.

Vector ruggedness measure

The `VRM` function calculates the vector ruggedness measure (VRM), which quantifies roughness based on the dispersion of unit vectors normal to the DTM surface. This measures the variability of both the magnitude and direction of slopes within a focal window by calculating the local (3 × 3 cell) slope and aspect and constructing unit vectors normal to each cell in the DTM (Figure 6) (Hobson, 1972; McKean & Roering, 2004; Sappington et al., 2007). Unit vectors are then decomposed into their corresponding *x*, *y*, and *z* components (i.e., the *x*, *y*, and *z* coordinates of the head of the vector relative to its origin) (Figure 6, Equations (2)–(4)). VRM is then calculated according to Equation (5) where *x*, *y*, and *z* are the vector components, and *N* is the number of cells in the window. This provides a value ranging from zero to one, representing completely smooth to rough surfaces, respectively.

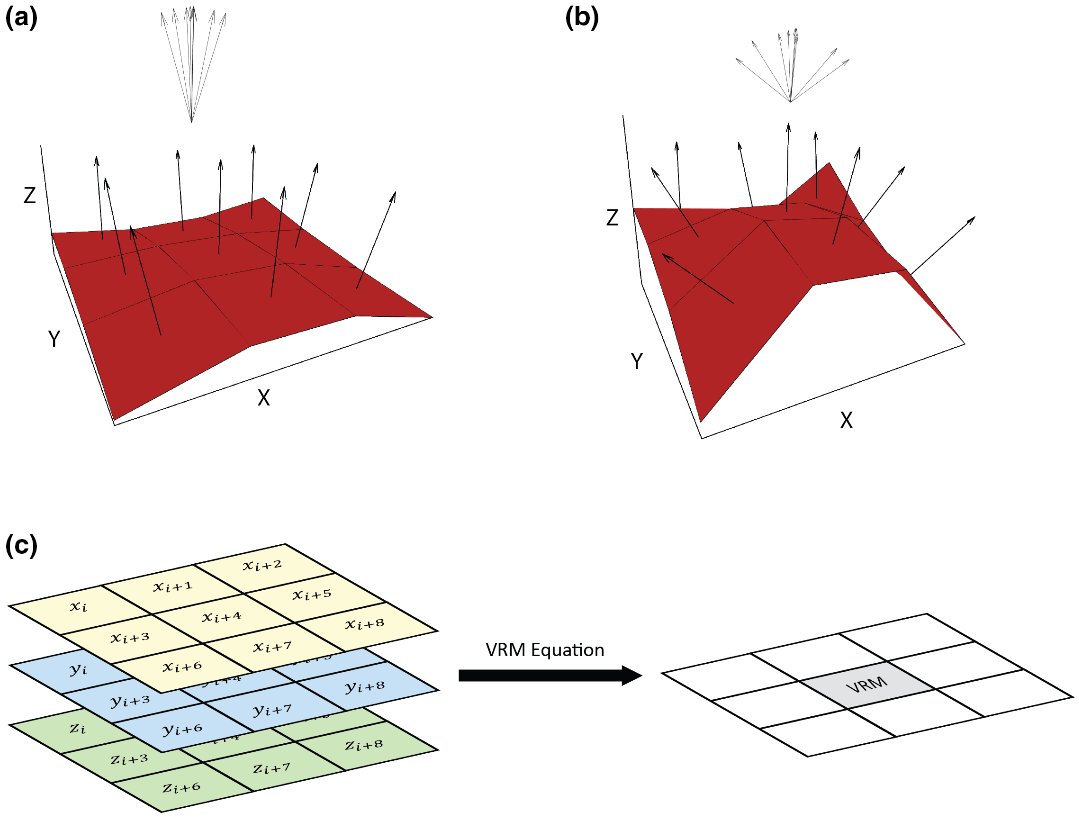


FIGURE 6 Vector ruggedness measure (VRM) measures the dispersion of unit vectors normal to each cell within the focal window. Smoother surfaces will have vectors with low dispersion (a) whereas rougher surfaces will have vectors with larger dispersion (b). These vectors are then decomposed into their *x*, *y*, and *z* components and used to calculate the VRM (c). Figure adapted from Sappington et al. (2007) and Habib (2021).

$$x = \sin(\text{slope}) * \sin(\text{aspect}) \tag{2}$$

$$y = \sin(\text{slope}) * \cos(\text{aspect}) \tag{3}$$

$$z = \cos(\text{slope}) \tag{4}$$

$$\text{VRM} = 1 - \frac{\sqrt{\left(\sum x\right)^2 + \left(\sum y\right)^2 + \left(\sum z\right)^2}}{N} \tag{5}$$

Surface area to planar area ratio

The `SAPA` function calculates the surface area to planar area ratio (SAPA) (Du Preez, 2015; Jenness, 2004). The values of SAPA range from one to infinity, where rougher surfaces are indicated by greater values as a result of increased surface area per unit planar area, and a completely smooth surface will have a value of one. This is a 3D analog to the classical "chain-and-tape" method, which calculates roughness as the ratio of the contoured distance (chain length) and linear distance (tape measure distance) (Figure 7) (Risk, 1972). Roughness measured in these ways is often referred to as "rugosity" (Friedman et al., 2012). The SAPA method first generates a raster image where the surface area of each cell is estimated by connecting a cell to its immediate neighbors to create eight triangles. These triangles are scaled down to the size of a single cell via the principle of similar triangles, and the areas of the triangles are summed (Figure 8) (Jenness, 2004). This is calculated within `SAPA` using the `SurfaceArea` function, which can also be called directly if only a raster of surface area is needed. This surface area is divided by the planar area to provide a measure of roughness. In the original measure proposed by Jenness (2004), this planar area was calculated as the product of the *x* and *y* cell resolutions, which implicitly assumes a flat topography. Du Preez (2015) improved this measure by correcting the planar area for local slope using the arc-chord ratio which adjusts the uncorrected planar area by dividing by the cosine of slope (Equation 6). The default behavior of the `SAPA` function is to apply this arc-chord correction. The method by Du Preez (2015) provides a measure of roughness at the per cell level and is corrected using the 3 × 3 slope according to Horn (1981). We have extended this measure by allowing application to a variable focal window size. Surface area is additive; the surface area values of the individual cells may be summed to calculate the total surface area within a focal window (Jenness, 2004). Planar area of the focal window is again calculated using Equation (6), but in this case, *x* and *y* distance are the size of the focal window in the *x* and *y* dimension rather than the cell resolution, and θ is the multiscale rather than 3 × 3 cell slope. Multiscale slope is calculated with the `SlpAsp` function using the "queen" case at a window size two cells greater ($w + 2$) than the focal window size (w) used to sum the surface area. This

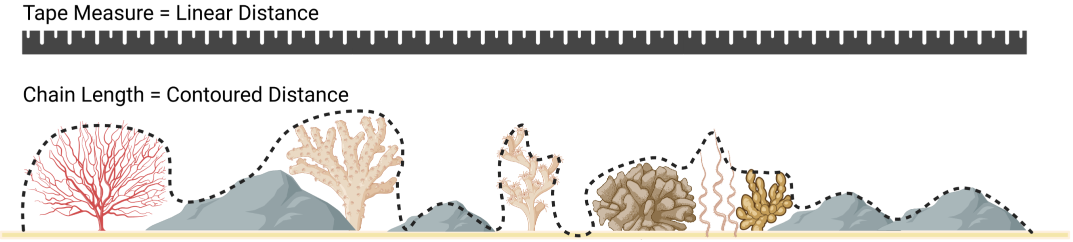

Tape Measure = Linear Distance
Chain Length = Contoured Distance

FIGURE 7 Roughness quantified using the "chain and tape" method which calculates roughness as the ratio of the linear distance (tape measure distance) to the contoured distance (chain length). Figure adapted from Friedman et al. (2012) and created with BioRender.com.

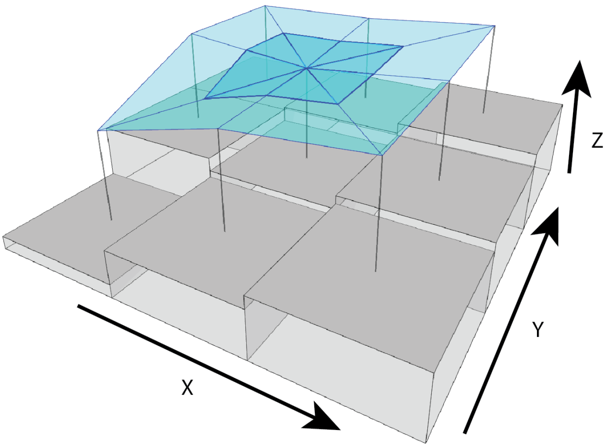


FIGURE 8 Schematic representation of how the surface area of each cell is calculated from a digital terrain model. First a central cell is connected to its surrounding neighbor cells to create eight large triangles. Then, these large triangles are trimmed back to the extent of the focal cell via the principle of similar triangles. Then the area of these eight smaller triangles is summed to provide an estimate of the surface area. Figure adapted from Jenness (2004).

larger window size is used for computing the slope since the plane of best fit should be based only on the "boundary data" which is the data immediately outside the main focal window (Du Preez, 2015). This implementation is a generalization of the Du Preez (2015) algorithm and is, therefore, equivalent for the per cell (i.e., 1 × 1 cell) case.

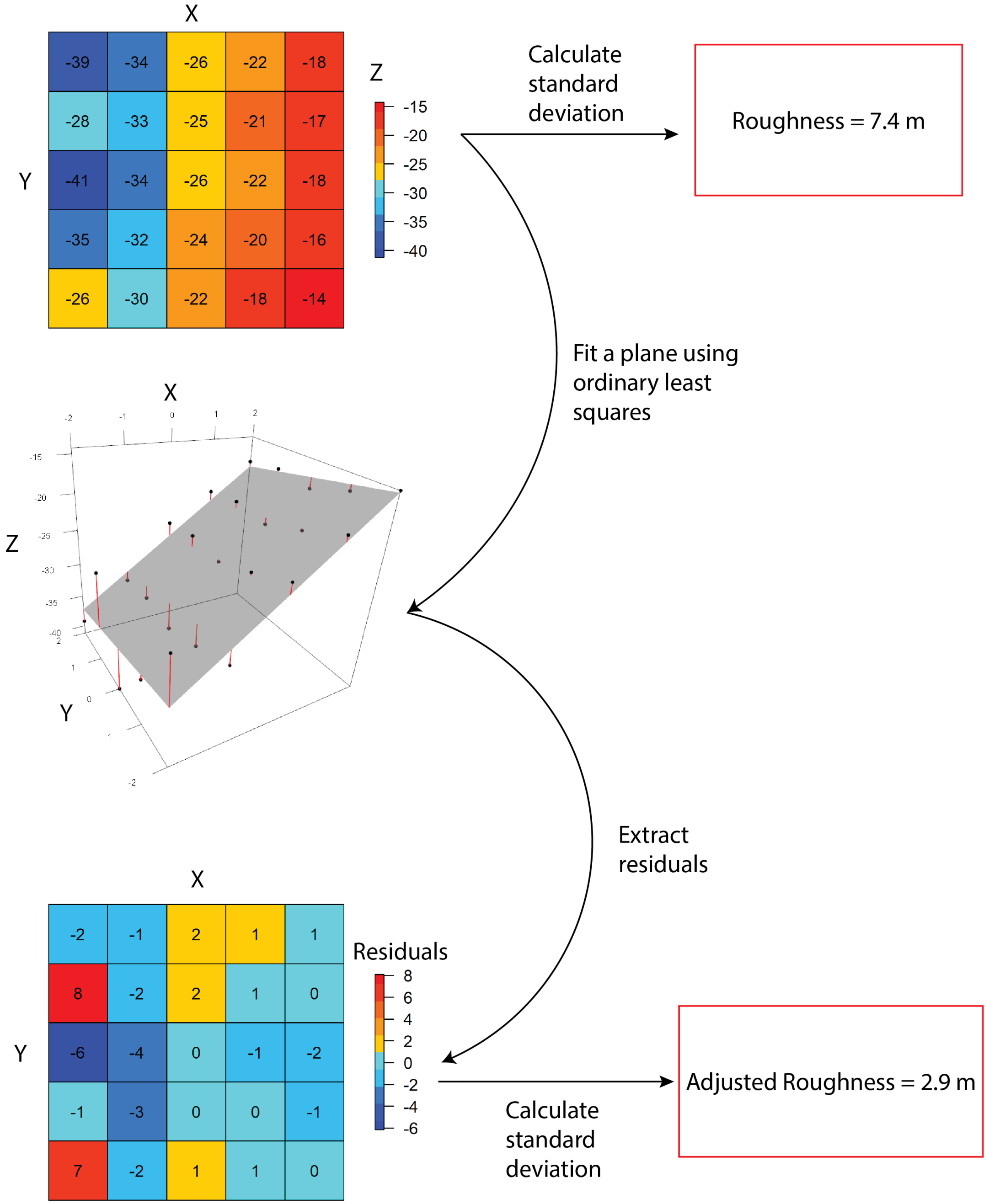
$$\text{Planar area} = \frac{x_{\text{dist}} y_{\text{dist}}}{\cos(\theta)} \tag{6}$$

Adjusted standard deviation

The `AdjSD` function calculates the local standard deviation of elevation or depth values in a focal window after removing the influence of slope. Local standard deviation is commonly used as a measure of roughness because it is easily calculated and scaled to multiple different window sizes (Lecours et al., 2017), yet even a perfectly smooth sloping surface will have non-zero standard deviation, which increases as the surface becomes steeper. Therefore, we propose a new roughness metric called the "adjusted standard deviation," which modifies the local standard deviation to account for the influence of slope. This is accomplished in three steps: (1) a plane is fit to the focal window using ordinary least squares where *Z* is modeled as a function of the *XY* coordinates in the focal window; (2) residuals between the plane and *Z* values are extracted, thus removing the effect of slope; and (3) the standard deviation of the residuals is calculated (Figure 9). The resulting measure is in the elevation units of the topographic surface and values can range from zero to infinity, with zero representing a perfectly smooth surface, regardless of its slope.

Roughness index-elevation

The `RIE` function calculates the roughness index-elevation, which conceptualizes roughness as the standard deviation of residual topography, much like adjusted standard deviation (Cavalli et al., 2008). Residual topography is calculated as the focal cell minus the focal mean. The local standard deviation is calculated from this residual topography using a focal filter (Figure 10). The use of the mean is appropriate since raster data is



-39	-34	-26	-22	-18
-28	-33	-25	-21	-17
-41	-34	-26	-22	-18
-35	-32	-24	-20	-16
-26	-30	-22	-18	-14

-2	-1	2	1	1
8	-2	2	1	0
-6	-4	0	-1	-2
-1	-3	0	0	-1
7	-2	1	1	0

X
Y
Z
-15
-20
-25
-30
-35
-40
Calculate standard deviation
Roughness = 7.4 m
Fit a plane using ordinary least squares
X
Z
Y
-2
-1
0
1
2
-15
-20
-25
-30
-35
-40
2
1
0
-1
-2
Extract residuals
X
Y
Residuals
8
6
4
2
0
-2
-4
-6
Calculate standard deviation
Adjusted Roughness = 2.9 m

FIGURE 9 Schematic representation how to compute roughness using adjusted standard deviation to account for the influence of slope contrasted with the traditional uncorrected local standard deviation.

regularly gridded and the focal cell is located at *x* and *y* coordinates of zero, so that the mean is exactly equal to the predicted value of the intercept of the plane. While conceptually similar to adjusted standard deviation, these methods are not equivalent. Roughness index-elevation calculates focal standard deviation using a raster of residual topography that is based on residuals calculated from a focal mean, for each cell. Adjusted standard

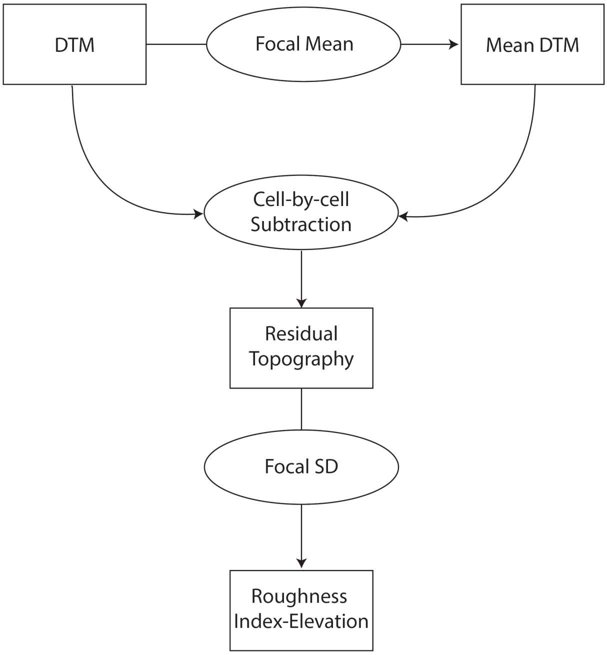


FIGURE 10 Steps for how to calculate the roughness index-elevation. Ovals represent operations and boxes represent raster surfaces. Figure adapted from Cavalli et al. (2008).

deviation extracts the residuals based on the fitted plane for the entire window and calculates the standard deviation on those values.

2.2 | Design and implementation

2.2.1 | Dependencies

The `MultiscaleDTM` package requires the use of the `terra` package (Hijmans, 2020) for the handling of spatial raster data, but is also backwards compatible with objects from the older `raster` package (Hijmans, 2010). The `terra` package is preferred here for terrain attribute calculation as it was designed as a successor to raster, has fewer dependencies and is faster since it is largely built in C++ (Stroustrup, 2013). Additionally, for increased speed with more complicated focal functions, `MultiscaleDTM` uses the `Rcpp` (Eddelbuettel et al., 2011) and `RcppArmadillo` (Eddelbuettel & Sanderson, 2014) R packages, which provide integration between R, C++, and the Armadillo C++ library for linear algebra and scientific computing (Sanderson & Curtin, 2016, 2020).

2.2.2 | Basic arguments

The `MultiscaleDTM` package was designed such that all terrain attributes can be calculated using a specified window size rather than restricting the user to a single default window size (usually 3×3 cells). Additionally, the functions are all designed in a consistent manner so that it is simple to calculate all available terrain attributes and pass them on to further analyses once the user learns the basic syntax. The main arguments that are shared across many functions are:

- `r`: A DTM raster.
- `w`: For rectangular shaped focal windows, a vector or single number representing the size of the focal window in terms or rows and columns. For functions that support other window shapes, `w` is the radius for circle shaped windows, the inner and outer radius for annulus shaped focal windows, or it can be a matrix indicating which cells to include and exclude in calculations.
- `na.rm`: Whether or not to remove `NA` (i.e., missing) values prior to conducting calculations.
- `metrics`: For functions that calculate several terrain attributes, this provides the ability to only return measures requested by the user.
- `include_scale`: Whether to append the scale (e.g., “5×5”) to the layer name. This can be useful when creating multilayer rasters with the same terrain attribute at multiple spatial scales.

2.2.3 | Data assumptions and design decisions

For calculations to be correct, some assumptions are made regarding the underlying data. First, for the *Z* axis (elevation), positive indicates up and negative indicates down. Therefore, when using a bathymetry surface, depth should be indicated using negative values (e.g., 45 m depth should be stored as −45 in the DTM). Additionally, for the *Y* axis, “up” is assumed to represent due North, which will not be true if a rotated coordinate system or azimuthal projection is used. This is particularly important when calculating aspect. Moreover, for slope, aspect, curvature, and some roughness measures, data must be in a projected coordinate system (not latitude and longitude; Table 2) where elevation units match map units (e.g., it is improper to have *X*/*Y* coordinates in meters and elevation in feet). Note, however, that slope and aspect can be calculated for unprojected data using a 3×3 cell window size using the `terrain` function of the `terra` package, which uses the geodetic formulas from Karney (2013). For unprojected data, the `terrain` function assumes that the Z units are meters, and this is also assumed by the `VRM` function from `MultiscaleDTM`, which uses `terrain` internally.

In all metrics where the standard deviation is calculated, the equation for the sample standard deviation is used which uses a denominator of $N-1$ rather than N, where N is the number of samples. Additionally, for functions that calculate aspect, if slope is zero, the default behavior is for aspect to return an NA value, and for eastness and northness to both return zero. Lastly, if all regression parameters other than the intercept evaluate to zero (i.e., the fitted surface is a perfectly flat and planar), the formulas for planform, profile, and twisting curvature in a strict sense would return `NA` (Appendix A); however, since these surfaces are entirely flat and planar and, therefore, have no curvature, these curvatures are instead returned as zero for these cases.

3 | `MultiscaleDTM` DEMONSTRATION

Several of the functions and capabilities of the `MultiscaleDTM` package are demonstrated below. Here, we focus on: (1) the `Qfit` function for its ability to calculate slope, aspect, and curvature measures all at once; (2) `AdjSD`, since it is a newly proposed measure of roughness; and (3) `RelPos` since it is a flexible method of calculating relative position. For more information, the function help documentation can be found using the “?” operator followed by the function name, or `help(package=“MultiscaleDTM”)`, which lists all available functions. Additionally, up-to-date documentation and tutorials can be found online at https://ailich.github.io/MultiscaleDTM/ or accessed from within R using `browseVignettes(“MultiscaleDTM”)`.

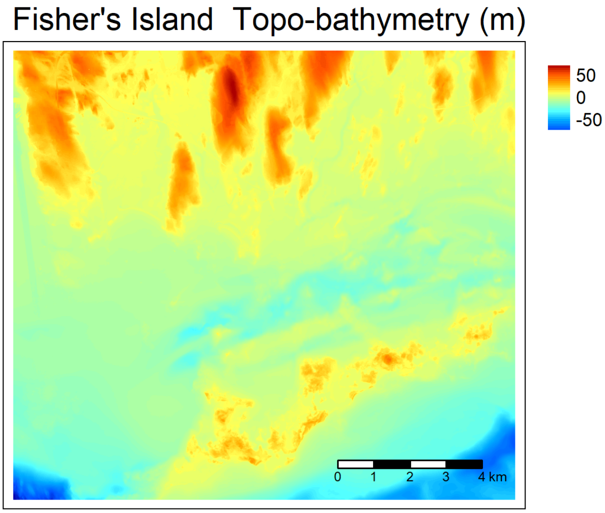


FIGURE 11 A 3 m resolution topo-bathymetry digital terrain model surface of Fisher's Island, NY, from the NOAA NCEI continuously updated digital elevation model.

3.1 | Data

The surface used for demonstrations of the various capabilities of the package is a 3 m resolution topo-bathymetry surface of Fisher's Island, New York, USA, and the surrounding seafloor (Figure 11). Fisher's Island is a small island (total area 173 km^2) at the eastern end of the Long Island Sound. The data were downloaded from the National Oceanic and Atmospheric Administration National Centers for Environmental Information (NOAA NCEI) Continuously Updated Digital Elevation Model ("Updated Digital Elevation Model (CUDEM)—1/9 Arc-Second Resolution Bathymetric-Topographic Tiles. [northeast_sandy]", NOAA NCEI, 2014).

3.2 | Getting started: Installation and loading the library

The first step is to install the `MultiscaleDTM` package which is available on both CRAN and github.

```
install.packages("MultiscaleDTM") #Install MultiscaleDTM
```

To load the `MultiscaleDTM` package, simply use the `library` function. This will also automatically load the required `terra` package which handles spatial data.

```
library(MultiscaleDTM)
```

3.3 | Reading in data

The `rast` function from the `terra` package is used to load spatial raster data into R (e.g., GeoTIFF, ascii, netCDF, GeoPackage, etc.). To load the Fisher's Island DTM, we supply the file path to this function.

```
DTM<- rast("Data/fishersisland_dtm.tif")
```

3.4 | Calculating slope, aspect, curvature, and morphometric features via quadratic surface fit

The Qfit function fits a quadratic surface to a focal window and can be used to calculate slope, aspect, curvatures, and provide a map of discrete landform classes known as morphometric features. Using Qfit can be as simple as calling the function on an object.

```
qfit1<- Qfit(DTM)
```

This will use the default values for all parameters, and will, therefore, return all available terrain attributes calculated using a 3×3 focal window, with slope and aspect reported in degrees. The returned terrain attributes are a smoothed version of elevation (“elev”), slope (“qslope”), aspect (“qaspect”), northness (“qnorthness”), eastness (“qeastness”), profile, planform, twisting, mean, max, and min curvatures (“profc,” “planc,” “twistc,” “meanc,” “maxc,” and “minc,” respectively), and morphometric features (“features”) (Figure 12). Note, the “*q*” preceding the slope, aspect, eastness, and northness names, which indicates the terrain attributes were calculated using Qfit, preventing name conflicts with SlpAsp which can calculate those terrain attributes using a different algorithm. Note that while both Qfit and SlpAsp can calculate these terrain attributes based on first-order derivatives, only Qfit can calculate curvatures, which are based on second order derivatives.

Although Qfit provides sensible defaults, in many cases it is necessary to change those parameters. Due to the large number of measures available and the flexibility of the method itself, Qfit has more parameters than any other function in the package. Three of the most useful parameters are w, which indicates the window size, metrics, which allows the user to specify which measures should be returned, and na.rm, which allows for the removal of NA values prior to calculations. The include_scale parameter can also be set to TRUE in order to append the window size to the layer name. Some additional options in Qfit allow for forcing the quadratic through the focal cell and altering the slope and curvature thresholds for determining morphometric features. For example, a subset of measures could be calculated using a focal window size of 17×17 where the surface is forced through the central cell, morphometric features are determined using a two degree slope threshold instead of the default one degree, scale is appended to the layer names, and NA values are removed prior to calculation. A comparison between select terrain attributes calculated using the 3×3 cell and 17×17 cell window is provided in Figure 13. Lastly, Qfit can return regression parameters or the partial derivatives allowing for calculation of alternative terrain attributes based on surface derivatives such as those based on equations contained within Wood (1996) and Minár et al. (2020) (Appendix C).

```
qfit2<- Qfit(DTM, w = 17, metrics = c("qslope", "qaspect",
                          "qeastness", "qnorthness",
                          "profc", "planc", "twistc",
                          "meanc", "features"),
             slope_tolerance = 2, force_center = TRUE,
             include_scale = TRUE, na.rm=TRUE)
```

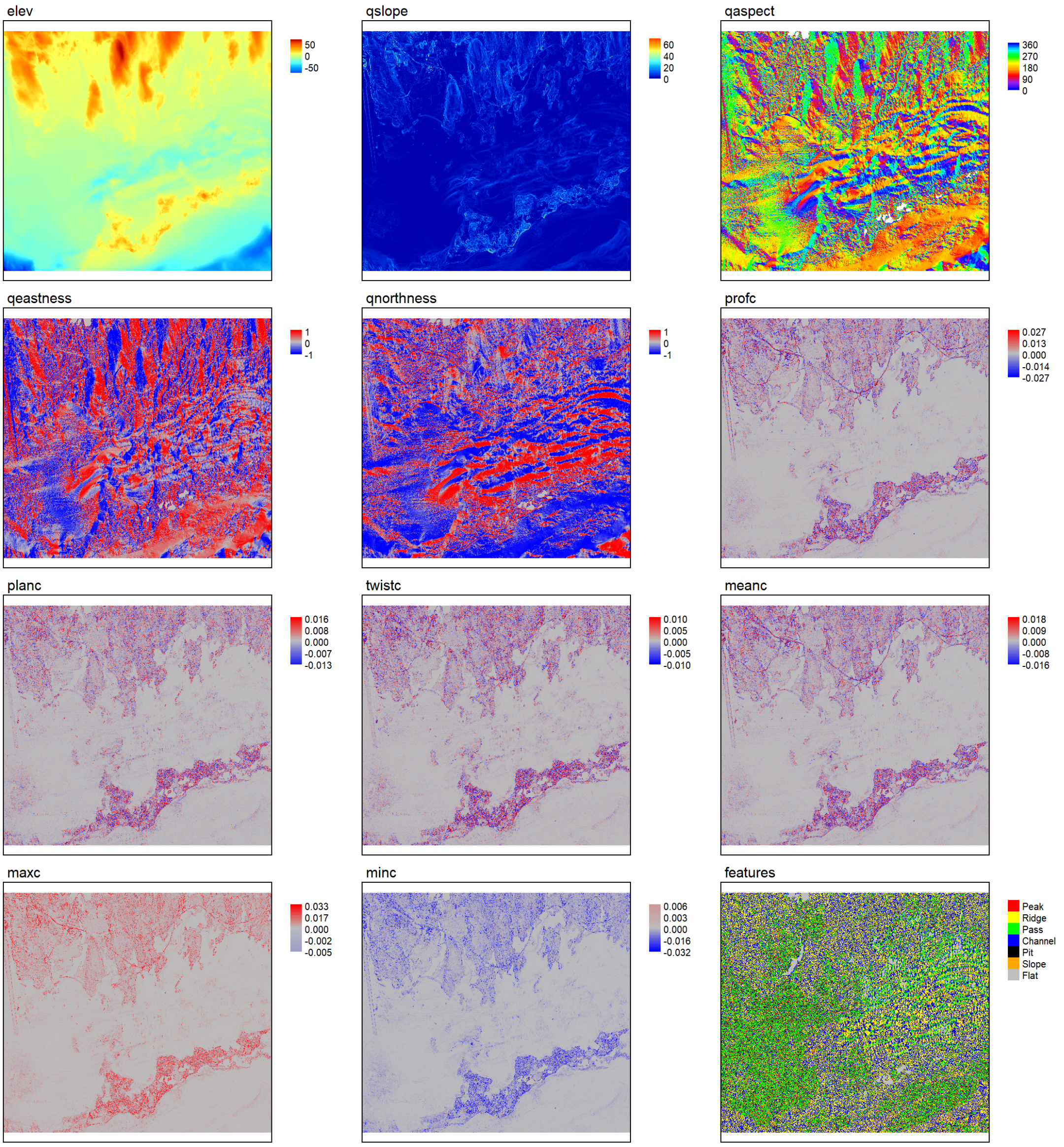


FIGURE 12 Plot of terrain attributes calculated with `Qfit` on the Fisher's Island digital terrain model using the default parameters.

3.5 | Calculating roughness via adjusted standard deviation

Adjusted standard deviation can be calculated using the `AdjSD` function. Again, the default parameters could be used by simply supplying only the DTM to the function as shown below.

```
adjsd1<- AdjSD(DTM)
```

However, just like `Qfit` and all other functions in the `MultiscaleDTM` package, the spatial scale can be changed. Additionally, while it is common to use a square focal window, a rectangular focal window can be specified by using a

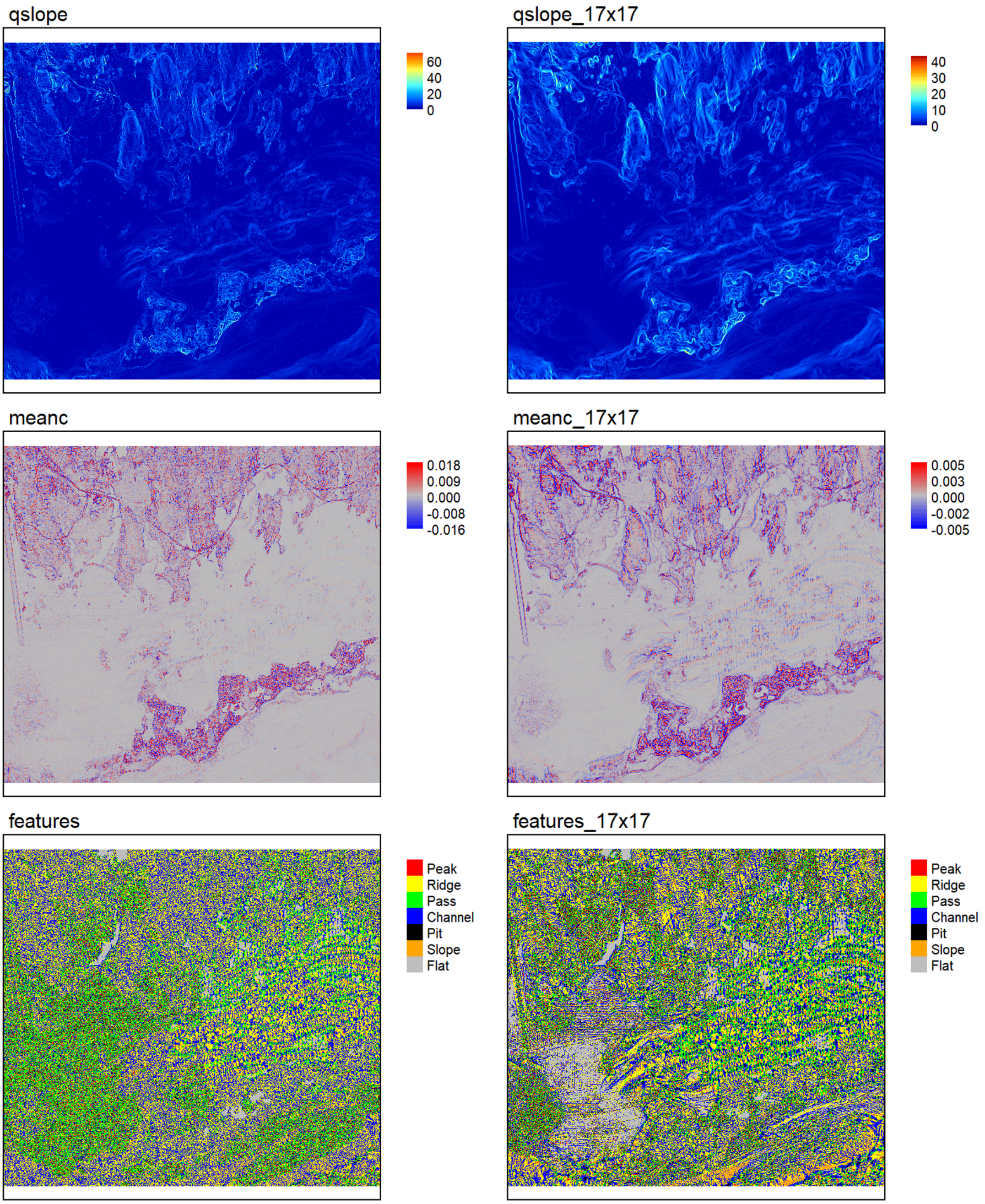


FIGURE 13 Comparison of select terrain attributes calculated with `Qfit` on the Fisher's Island digital terrain model using a 3×3 cell focal window (left) and a 17×17 cell focal window (right).

vector of two numbers indicating the rows and columns (in that order) of the focal window. The following code could be used to calculate adjusted standard deviation with a focal window of seven rows and nine columns. Note that similar to `Qfit`, `NA` values can be removed prior to computations and the window size can be appended to the layer name to indicate the scale (Figure 14).

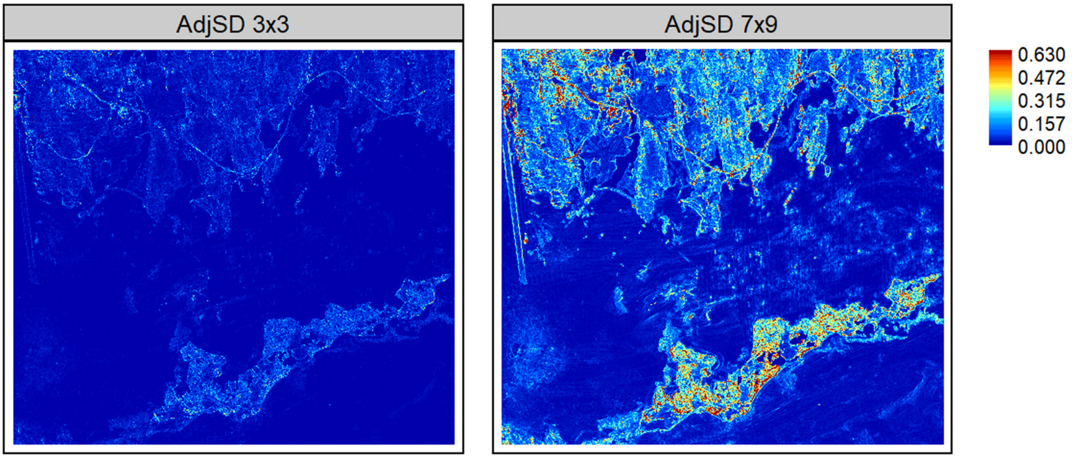


FIGURE 14 Adjusted standard deviation of Fisher's Island digital terrain model using a 3×3 and 7×9 cell focal window.

```
    adjsd2<- AdjSD(DTM, w= c(7,9),
                   na.rm = TRUE, include_scale = TRUE)
```

3.6 | Calculating relative position

Relative position can be calculated with the `RelPos` function. This function currently supports rectangular, circular, annulus, and custom shaped focal windows (Table 2). We will use an annulus shaped window in this example, which requires the specification of an inner and outer radius. This inner and outer radius is supplied to the `w` parameter and can be in "cell" or "map" units (e.g., meters). Additionally, we will explicitly specify to use the mean of the included values in the focal window as the reference height and not to standardize values based on local topography using the `fun` and `stand` parameters, respectively. The following code could be run to calculate relative position using an inner radius of four cells and an outer radius of six cells just as in the example annulus window shown in Figure 3.

```
  rp<- RelPos(DTM, w=c(4,6), shape="annulus", stand="none",
  unit="cell", fun="mean")
```

Or equivalently, since the DTM has 3 m resolution, the window could be specified as 12 and 18 m for the inner and outer radius, respectively.

```
rp<- RelPos(DTM, w=c(12,18), shape="annulus", stand="none", unit="map", fun="mean")
```

Moreover, the `annulus_window` function can be used to check that the annulus was properly specified. Ones indicate that the value is included in the annulus and `NA` indicates that it is excluded.

```
annulus_window(radius = c(4,6), unit = "cell")
```

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13]
 [1,]   NA   NA   NA   NA   NA   NA    1   NA   NA    NA    NA    NA    NA
 [2,]   NA   NA   NA    1    1    1    1    1    1     1    NA    NA    NA
 [3,]   NA   NA    1    1    1    1    1    1    1     1     1    NA    NA
 [4,]   NA    1    1    1   NA   NA   NA   NA   NA     1     1     1    NA
 [5,]   NA    1    1   NA   NA   NA   NA   NA   NA    NA     1     1    NA
 [6,]   NA    1    1   NA   NA   NA   NA   NA   NA    NA     1     1    NA
 [7,]    1    1    1   NA   NA   NA   NA   NA   NA    NA     1     1     1
 [8,]   NA    1    1   NA   NA   NA   NA   NA   NA    NA     1     1    NA
 [9,]   NA    1    1   NA   NA   NA   NA   NA   NA    NA     1     1    NA
[10,]   NA    1    1    1   NA   NA   NA   NA   NA     1     1     1    NA
[11,]   NA   NA    1    1    1    1    1    1    1     1     1    NA    NA
[12,]   NA   NA   NA    1    1    1    1    1    1     1    NA    NA    NA
[13,]   NA   NA   NA   NA   NA   NA    1   NA   NA    NA    NA    NA    NA
attr(,"unit")
[1] "cell"
attr(,"scale")
[1] "4x6"
attr(,"shape")
[1] "annulus"
```

The matrix resulting from this function can be directly passed to the `w` parameter instead of specifying a radius and units in `RelPos`. Both of the options below options provide the same result as the code above that specified `w` using a vector containing the inner and outer radii.

```
rp<- RelPos(DTM, w=annulus_window(radius=c(4,6),unit="cell"),
            shape="annulus", stand="none", fun="mean")
rp<- RelPos(DTM, w=annulus_window(radius=c(12,18),unit="map",res=3),
            shape="annulus", stand="none", fun="mean")
```

The resultant relative position surface can be seen in Figure 15. Relative position is measured in the units of the DTM (meters) where negative values represent local topographic lows (i.e., depressions), and positive values

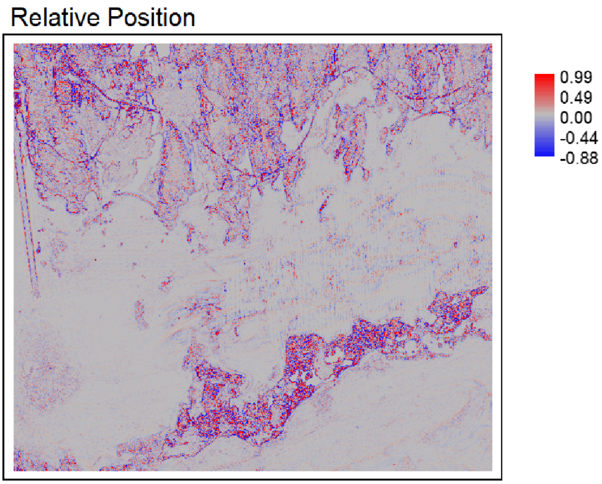


FIGURE 15 Relative position calculated using an annulus window with an inner radius of four cells (12 m) and an outer radius of six cells (18 m).

represent local highs (i.e., peaks). Although not shown here, `RelPos` like most other functions in the package supports the parameters `na.rm` and `include_scale`. Note if `include_scale` was set to be true, for this example, since an annulus window is used, the the inner and outer radii (in cell or map units depending on what was specified) rather than the number of rows and columns in a rectangular window would be appended to the layer name (e.g., "rpos_4x6" if cell units are specified or "rpos_12MUx18MU" if map units are specified).

3.7 | Further analyses

It is outside the scope of this article to explore the multitude of analyses that could be conducted using terrain attributes, but a wide array of tools available in R could be applied for various applications. Simple classification dictionary-based approaches, such as those available in the ArcGIS Benthic Terrain Modeler (Lundblad et al., 2006; Walbridge et al., 2018) can be accomplished using the `lapp` function in `terra` in combination with conditional statements (Appendix C). Moreover, R has advanced built-in statistical capabilities (e.g., generalized linear models), and more sophisticated statistical and machine learning techniques are available in packages such as `caret` (Kuhn, 2008), `parsnip` (Kuhn & Vaughan, 2022), and `mlr3` (Lang et al., 2019), among others. Many of these can be used to make spatial predictions across the domain of the spatial raster data thanks to the extensible `predict` function in `terra`, which supports a wide variety of model types. Additionally, the `Rstoolbox` package (Leutner & Horning, 2016) is built specifically for use with remote sensing data and can simplify workflows involving unsupervised and supervised classification procedures, which are common in remote sensing, ecology, and habitat mapping (Brown et al., 2011). Finally, geographic object-based image analysis and spatial segmentation can be conducted with the `SegOptim` (Gonçalves et al., 2019) and `supercells` (Nowosad & Stepinski, 2021) packages.

3.8 | Exporting data

All surfaces can be written to a file (e.g., GeoTIFF, ascii, netCDF, GeoPackage, etc.) using either the `writeRaster` function in the `terra` package or by supplying a file name to the `filename` argument in any of the

MultiscaleDTM functions that return terrain attributes, therefore ensuring interoperability with any other GIS software.

4 | ROUGHNESS SIMULATION

4.1 | Data

Artificial surfaces are useful for testing hypotheses in a controlled environment over a wide range of differing conditions (Halley et al., 2004; Lecours et al., 2017). Three artificial surfaces of various complexity are used here for simulation (Figure 16). These surfaces are from Lecours et al. (2017) and consist of simulated landscapes of low, medium, and high complexity (fractal dimensions of 2.06, 2.40, and 2.79, respectively). Additionally, the Fisher's Island digital terrain model from the demonstration (Figure 11) is used as a "real-world" case study in this simulation.

4.2 | Methods

Simulation was used to examine the Spearman rank correlation (ρ) between slope and the different roughness measures included in the software (adjusted standard deviation, slope corrected SAPA, VRM, and roughness index-elevation; Table 1). Additionally, the standard deviation of depth values (without adjustment) was calculated as a control to examine the effect of slope on uncorrected roughness measures. This simulation was conducted using the three artificial (Figure 16) and one real-world (Figure 11) dataset. For the real world surface, a representative subset of the data was used to reduce computational intensity by drawing 10,000 random samples without replacement. Additionally, to determine whether results were dependent on the analytical spatial scale, tests were conducted at both fine and broad scales by varying the window size at which the attributes were calculated. For the fine-scale analysis, all terrain attributes were calculated using a 3 × 3 cell focal window except for SAPA, which was calculated on a per cell basis (i.e., 1 × 1 cell). For the broad-scale analysis, all terrain attributes were calculated using a 17 × 17 cell focal window except for SAPA, which was calculated using a 15 × 15 cell window. The smaller window size was used for SAPA since the slope correction is based on the boundary data immediately outside of the focal window (Du Preez, 2015), making it appropriate to compare SAPA to the slope at the scale being used to adjust the planar area.

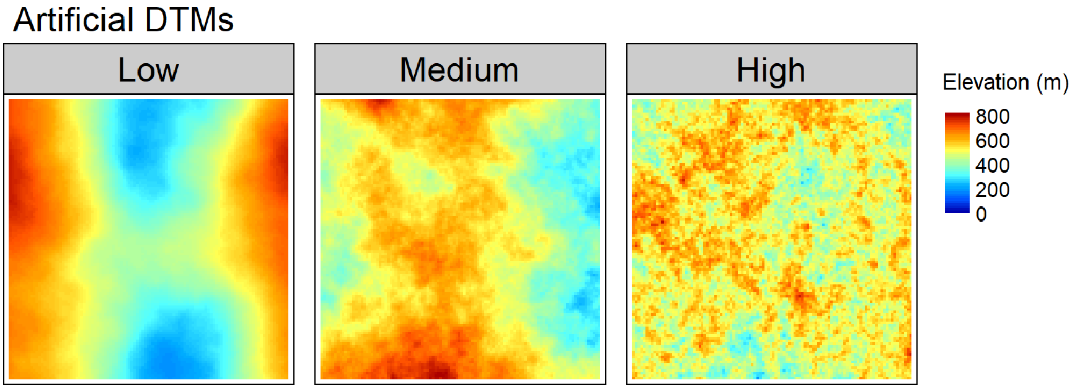


FIGURE 16 Digital terrain models of low, medium, and high complexity artifical landscapes (generated using fractal dimensions of 2.06, 2.40, and 2.79, respectively).

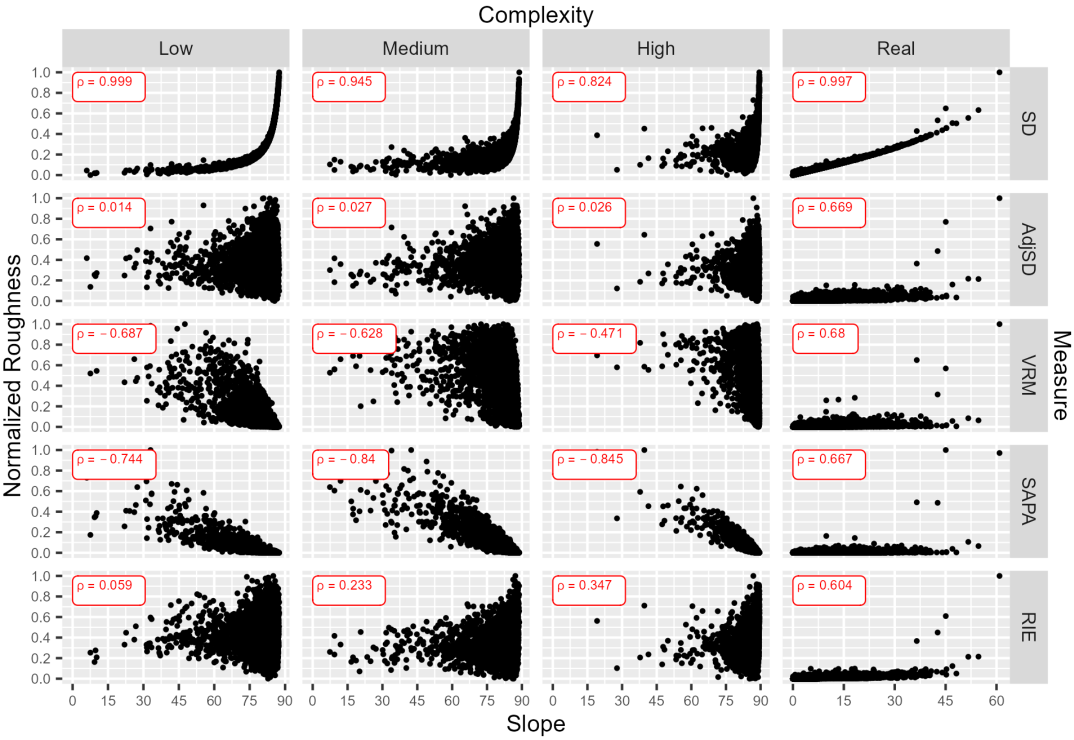


FIGURE 17 Scatter plots showing the results from the fine-scale roughness simulation. The *x* axis represents slope in degrees, and the *y* axis represents normalized roughness. Since each roughness measure has a different scale and the range of values differ by data complexity, to facilitate comparison, values on the *y* axis have been normalized to a range of zero to one using the equation Normalized Roughness = (value − max)/(max − min). Columns represent the data surface: low complexity, medium complexity, high complexity, and real-world surface. Rows represent the roughness measure: Standard deviation ("SD"), adjusted standard deviation ("AdjSD"), vector ruggedness measure ("VRM"), slope corrected surface area to planar area ratio ("SAPA"), and roughness index-elevation ("RIE").

4.3 | Results

The results from the roughness simulation at fine and broad scales are shown in Figures 17 and 18, respectively. The traditional standard deviation method generally shows the highest magnitude rank correlation. It had the highest value in six of the eight scenarios, and had $|\rho| \geq 0.8$ in all cases except for one. The strength of this relationship increased with decreasing complexity and a finer spatial scale. Additionally, the relationship between standard deviation and slope was always positive. VRM and SAPA, however, displayed substantial negative correlations with slope in all of the artificial datasets, with the ρ ranging from −0.880 to −0.471 for VRM, and −0.994 to −0.744 for SAPA. Adjusted standard deviation performed best for all artificial datasets in both the fine and broad scales with the lowest $|\rho|$ in all six of those scenarios. Often $|\rho|$ was near zero for this measure and $|\rho|$ was >0.1 in only one of the six scenarios. Roughness index-elevation showed similar patterns in the scatter plots to adjusted standard deviation and was the second best performing measure across all the artificial datasets with $|\rho|$ ranging from 0.059 to 0.347. In all cases, roughness index-elevation displayed a positive relationship with slope, and adjusted standard deviation had a positive relationship in all except for the low-complexity broad-scale simulation where a weak negative relationship was observed ($\rho = -0.135$). Lastly, roughness index-elevation was the best performing measure for the real world dataset in both the fine and broad-scale analyses; however, this difference was relatively small and all methods that account for slope performed similarly with $|\rho|$ values of about

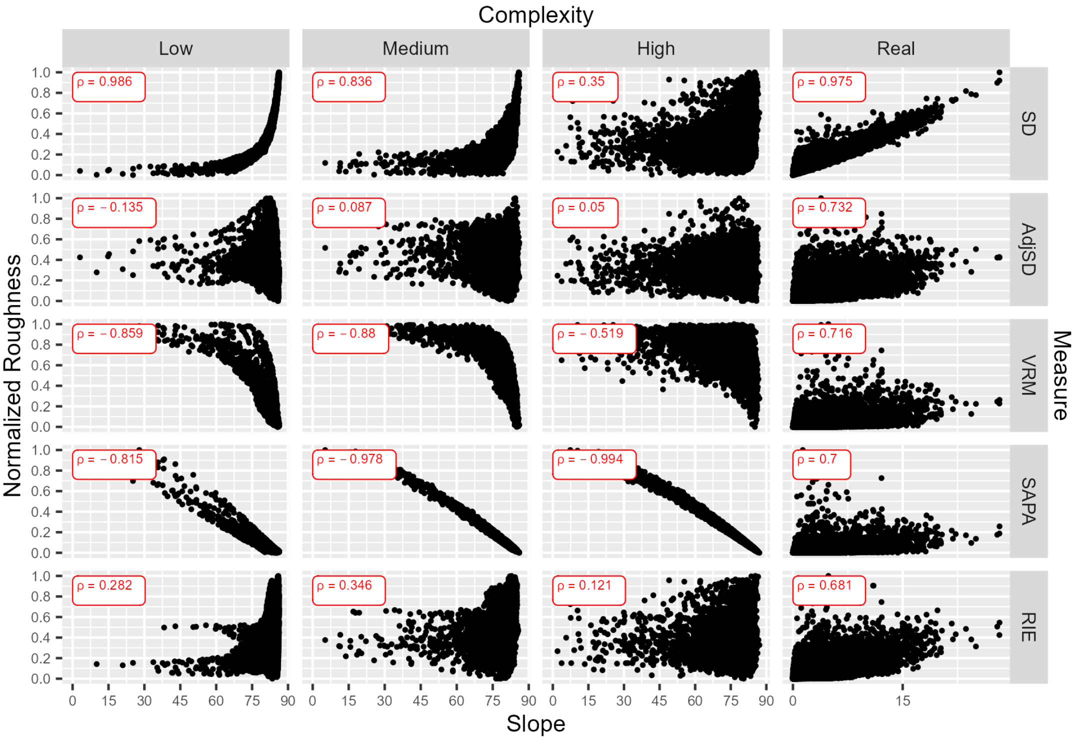


FIGURE 18 Scatter plots showing the results from the broad scale roughness simulation. The *x* axis represents slope in degrees, and the *y* axis represents normalized roughness. Since each roughness measure has a different scale and the range of values differ by data complexity, to facilitate comparison, values on the *y* axis have been normalized to a range of zero to one using the equation Normalized Roughness = (value − max)/(max − min). Columns represent the data surface: low complexity, medium complexity, high complexity, and real-world surface. Rows represent the roughness measure: Standard deviation ("SD"), adjusted standard deviation ("AdjSD"), vector ruggedness measure ("VRM"), slope corrected surface area to planar area ratio ("SAPA"), and roughness index-elevation ("RIE").

0.6–0.7 compared with the uncorrected standard deviation which had $|\rho|$ values of 0.997 and 0.975 for the fine and broad-scale analyses, respectively.

5 | DISCUSSION

5.1 | MultiscaleDTM demonstration

Our demonstration shows how various terrain attributes can easily be calculated at multiple spatial scales using `MultiscaleDTM`. Differences can be seen in the surfaces calculated at different spatial scales. For example, when using `Qfit`, the range of slope and curvature values decreases as the spatial scale of analysis increases, and the broader-scale map of morphometric features has lower speckle and some larger continuous features become more apparent (Figure 13). Additionally, for the example using `AdjSD`, greater variability is detected at broader spatial scales (Figure 14). Using a larger focal window can smooth out artifacts in the DTM, reduce speckle, and reveal broader scale features that may not be apparent in the default 3 × 3 cell window. While multiscale capabilities increase the complexity of the analyses and decisions that need to be made, they also provide powerful tools to scientists. Often a 3 × 3 cell window is used to define the analytical spatial scale, even if it may not be the most appropriate option (Wang et al., 2010). For example, multiscale analyses provide the opportunity to more accurately represent the complexity and ambiguity in boundaries between landforms via fuzzy set theory (Fisher et al., 2005; Wang et al., 2010) and can be used to match the analytical scales to the operational or natural scales. This is essential for accurate prediction and inference in disciplines like geography, ecology, and geomorphology (Cavazzi et al., 2013; Goodchild, 2011; Lecours et al., 2015; Levin, 1992; Wiens, 1989). Whether attempting to determine relevant characteristic scale(s) of a specific dataset (Behrens et al., 2019; Drăguţ et al., 2011), use multiscale terrain attributes to maximize predictive accuracy (Behrens et al., 2010; Misiuk et al., 2018; Sîrbu et al., 2019), appropriately combine information across scales into a single measure (Dolan, 2012; Wang et al., 2010) or adapt the scale to the local environment (Lindsay et al., 2019; Shang et al., 2021), using multiscale terrain attributes most effectively is an area of ongoing research with evolving guidance (Maxwell & Shobe, 2022; Misiuk et al., 2021). While MultiscaleDTM does not explicitly solve this challenge, enabling transparent calculation of a wide array of terrain attributes at multiple scales can aid in conducting studies to further investigate these questions.

Common software packages used to calculate terrain attributes include QGIS (QGIS Development Team, 2009), GRASS GIS (Neteler et al., 2012), SAGA GIS (Conrad et al., 2015), GDAL (GDAL/OGR contributors, 2000), ArcGIS, ENVI, WhiteboxTools (Lindsay, 2016), LandSerf (Wood, 2009), and MicroDEM (Guth, 2009). While many of the algorithms presented here are available in these other software packages (both open-source and proprietary as well as free and paid software), we believe that MultiscaleDTM provides significant added value. It is not the goal of MultiscaleDTM to replace these software packages; instead, it is meant to simplify workflows for those who are already using or interested in performing geospatial analysis using R. Given the advanced capabilities and popularity of R within the spatial science community (Bivand, 2021; Lovelace et al., 2019), the lack of a dedicated R package for calculating multiscale terrain attributes was a clear gap. If a user is already familiar with R and particularly with using the terra and/or raster packages for spatial data analysis, the learning curve for using MultiscaleDTM should be quite low as MultiscaleDTM uses objects from these packages directly and employs syntax that should be familiar to users of those R packages. Additionally, a new user could install the needed packages, import their DTM, calculate terrain attributes, and export them for use in their GIS of choice using just a few lines of code that could be adapted from the tutorial. Moreover, there are clear benefits both in terms of convenience and computational reproducibility that stem from being able to calculate all of these measures in a single software environment.

Additionally, MultiscaleDTM was designed to be transparent and flexible. Ambiguity in terminology and the specific algorithm being used can lead to confusion as to what is actually being calculated, which can lead to different results depending on the software used (Bishop et al., 2012; Bouchet et al., 2015; Lecours et al., 2017; Minár et al., 2020; Sofia, 2020). To overcome this, substantial effort is dedicated to describing which algorithm is being used in each function and how those algorithms work, both in this article and the documentation for the software so that all of this information is explicit to the user. Careful consideration was paid to the selection of algorithms, particularly in regards to curvature and roughness. We include the ability to calculate twisting curvature which is not available in most GIS software packages despite being one of the basic types of curvature, and calculate true geometric curvatures rather than the common directional derivatives or imitation curvatures, which have lower interpretation potential (Minár et al., 2020). For roughness, there are a wide array of algorithms that could be used, but we only included measures that have some way to account for the influence of slope (Cavalli et al., 2008; Du Preez, 2015; Grohmann et al., 2011; Hobson, 1972; Smith, 2014; Wilson et al., 2007). Moreover, as there is no one true way to conceptualize roughness (Grohmann et al., 2011; Hobson, 1972; Smith, 2014; Wilson et al., 2007), we provide measures that conceptualize roughness in several different ways so that the user can consider which one is most appropriate for their analysis. Additionally, as mentioned earlier, many software packages often restrict users to a 3×3 cell focal window for certain calculations (Lecours et al., 2017; Misiuk et al., 2021). All algorithms in the MultiscaleDTM package support multiscale calculation via a variable focal window size (Table 2). Lastly, MultiscaleDTM provides some other unique features that are unavailable or uncommon in other software packages including the adjusted standard deviation, multiscale slope corrected SAPA, explicit handling of missing values, and the ability to return regression parameters from the quadratic fit which can then be used to calculate custom terrain attributes based on surface derivatives (Appendix C).

5.2 | Roughness simulation

The results of the simulations showed that adjusted standard deviation performed the best for all artificial surface simulations, and had comparable performance to other slope-corrected measures in the real world example. The next best performance was exhibited by roughness index-elevation, which had the second lowest magnitude rank correlation with slope in all the artificial datasets, and performed slightly better than all other slope corrected measures in both the fine and broad-scale simulations for the real world dataset. Additionally, we found that if slope is not accounted for in the roughness measure itself, then the two will be strongly related, particularly in areas of lower complexity. This is consistent with previous findings in the literature (Du Preez, 2015; Grohmann et al., 2011; Ironside et al., 2018; Sappington et al., 2007) and indicates that roughness measures that account for slope should generally be preferred. VRM and SAPA with arc-chord correction are largely considered in the literature to provide measures of roughness that are independent of slope; however, the simulation demonstrated that both VRM and slope-corrected SAPA can display strong negative correlations under certain conditions, though the effect was stronger for SAPA. This relationship has largely been overlooked in the literature but was recently noted for VRM (Trevisani et al., 2022). The results of our simulation agree with the results of Trevisani et al. (2022), and indicate that this effect is also present for slope corrected SAPA. The strength of this relationship varied with spatial scale and data complexity. For both VRM and SAPA this effect seemed to be more severe at broader spatial scales. Stronger negative correlations were generally observed as data complexity increased for SAPA, but the strength of negative correlation was generally reduced as data complexity increased for VRM. While this effect was present in the artificial surfaces, both SAPA and VRM had comparable performance to other slope-corrected roughness measures for the real world dataset, indicating that the effect may not be strong in a practical setting. The difference in trends seen between the artificial and real world datasets may be due to differences in the underlying distribution of slope values. The real world dataset consists largely of gently sloped areas with a few steeper areas, whereas the artificial datasets consist mainly of areas with steep slopes and a few areas of shallow slopes. The simulations also showed that for the real world dataset, all slope corrected methods successfully decoupled slope and roughness. While the rank correlation between slope and these slope corrected roughness measures was moderately high (about 0.6–0.7), this was substantially reduced relative to the raw standard deviation of elevations values, and the scatter plots show no clear relationship (Figures 17 and 18).

In addition to reducing correlation with slope, there are also practical and conceptual considerations regarding the appropriateness of roughness measures. For example, measures that conceptualize roughness as residual topography (adjusted standard deviation and roughness index-elevation) express roughness in the units of the DTM, making its physical meaning easy to comprehend. This facilitates simulation of surfaces with varying levels of roughness (e.g., by drawing random values from a normal distribution with a specified standard deviation). Moreover, adjusted standard deviation is grounded mathematically through the application of ordinary least squares to detrend the surface, removing the effect of slope. Although roughness index-elevation does not use ordinary least squares, the mean is exactly equal to the predicted value of the intercept of the plane since the raster data is regularly gridded and the focal cell is located at *x* and *y* coordinates of zero. Since roughness index-elevation does not need to fit a plane to the data, it is computationally less demanding and additionally can be used for unprojected data in cases where it is reasonable to assume that the distance in Cartesian units (e.g., meters) between cells in the longitudinal direction does not vary substantially within the focal window. Therefore, it may be a suitable replacement for adjusted standard deviation in certain cases. VRM may be more difficult to conceptualize since it is a function of the magnitude and direction of decomposed unit vectors normal to the terrain surface. It is also the only measure that has an upper limit on roughness. Nevertheless, this method has a long established history in geomorphology, having been first proposed by Hobson (1972). Moreover, this measure explicitly accounts for unprojected (i.e., lat/lon) datasets by using geodesic distances (Karney, 2013). Lastly, SAPA is highly similar to how roughness is measured in the field (e.g., via diver surveys underwater) and, therefore, may provide intuitive meaning (Figure 7). Additionally, surface area is an important habitat component for sessile organisms by providing attachments surfaces and for mobile organisms by providing refugia from predators, making the surface area itself potentially relevant for ecological investigations. SAPA, however, is strongly related to the resolution of the raster. Fine-scale DTMs are capable of detecting more surface area than broader-scale DTMs. In fact, SAPA values may approach infinity as the resolution gets finer since the level of detail detected and, therefore, the surface area will increase (Mandelbrot, 1967; Smith, 2014).

5.3 | Future work and limitations

While this software should be a useful tool to scientists looking to calculate terrain attributes, there are both limitations and room for growth. While varying the focal window size is a convenient and intuitive method of changing the spatial scale, the processing time grows exponentially with window size (Grohmann & Riccomini, 2009; Newman et al., 2022). Additionally, analysis scales must be specified at discrete intervals that are odd integer multiples of the cell resolution (Newman et al., 2022). Spectral-based approaches such as wavelet decomposition (Csillag & Kabos, 2002) and Gaussian scale space (Behrens et al., 2018; Newman et al., 2022) show great promise in their ability to represent scale in a continuous fashion (Newman et al., 2022) and can scale better computationally, particularly with advances in computer vision (Kovesi, 2009). We note that the `MultiscaleDTM` package could be used alongside such alternative approaches. For example, `MultiscaleDTM` could be used to calculate terrain attributes using a 3×3 cell focal window, followed by the application of spectral analysis methods to vary the spatial scale by modifying the DTM prior to calculation of terrain attributes or modifying the attributes after calculation (Misiuk et al., 2021; Newman et al., 2022). Moreover, while computationally expensive, the computation time of focal operations can be improved via parallel computing (Zhang et al., 2021).

Several features could be added to the `MultiscaleDTM` package in the future. A logical first step would be to provide syntax to allow for integrated multiscale terrain attributes, which incorporate information across multiple spatial scales (Dolan, 2012; Misiuk et al., 2021; Wang et al., 2010). It is worth noting that these can already be determined using `MultiscaleDTM` by calculating a terrain attribute across several successively larger window sizes and then averaging the results with the desired weighting scheme. This could, however, be streamlined by providing support for a list of multiple spatial scales in the `w` argument and then automatically averaging the resultant surfaces within the function according to a user specified weighting scheme. Another future improvement would allow for cell weights to be used in focal functions. Particularly in `Qfit`, a weighted least squares approach could be used so that cells further from the focal cell have less influence (Albani et al., 2004; Wood, 1996). Additionally, while square windows are common, other shapes can also be useful (Lindsay et al., 2015; Schmidt & Andrew, 2005). Currently most functions support a rectangular focal window, which can be useful for specifying the window as a square in map units (e.g., meters) in cases where the *x* and *y* resolution are unequal, or for addressing anisotropy (Lindsay et al., 2015; Schmidt & Andrew, 2005). Much like in `RelPos`, future updates may provide support for a greater variety of focal window shapes across more functions (Table 2). Finally, greater support for unprojected data would aid in computing terrain attributes at a global scale. Currently only measures of relative position, roughness index-elevation, and VRM can be calculated on unprojected data when using `MultiscaleDTM` (Table 2). This prevents calculation of slope, aspect, curvature, and some forms of roughness for unprojected data, and, therefore, requires DTMs to be reprojected prior to conducting certain calculations. At best, this is suboptimal since projection alters properties of the data (Guth & Kane, 2021) and is not possible at a global extent. The algorithms used for planar data cannot be applied to unprojected data due to inherently different geometries of the surface (planar vs spheroid); however, algorithms do exist to measure slope, aspect, and curvature for unprojected data (Florinsky, 2017; Guth & Kane, 2021) via the use of geodetic formulas (Karney, 2013; Vincenty, 1975). These may be implemented in the future.

6 | CONCLUSIONS

Having the tools necessary to perform complete (geo)morphometric analyses in a single software package with consistent syntax streamlines workflows reduces the potential for errors. Geomorphometry is a rapidly evolving field, and the importance of better incorporating open science practices in the field has been highlighted (Sofia, 2020). Open science makes workflows reproducible and allows for a better understanding of the theories behind the tools, thus enabling disciplinary growth that is based on more robust theoretical foundations. By providing an open-source package with clear documentation describing its application and underlying theory, we aim to contribute to the development of open-science geomorphometry solutions.

`MultiscaleDTM` is designed to be free, transparent, flexible, scientifically rigorous, and easy to use. It can be applied to calculate members from each of the five thematic groups of terrain attributes (slope, aspect, curvature, relative position, and roughness) at multiple spatial scales of analysis with a consistent syntax. Additionally, the importance of choosing a roughness measure that explicitly accounts for the effect of slope was demonstrated, and the newly proposed adjusted standard deviation method is an effective solution. Finally, as an open-source package, we hope to receive feedback and contributions that facilitate continued development; volunteered contributions for optimization of existing functions or for new features are particularly encouraged.

ACKNOWLEDGMENTS

We would like to thank Dr. Robert Hijmans for adding functionality to the `terra` package specifically to facilitate the development of this R package, and for promptly responding to questions. Additionally, we would like to thank Dr. Jozef Minár for answering questions regarding curvature. Moreover, we would like to thank Dr. Don Chambers for providing insight into underlying mathematical assumptions of certain terrain attributes. We would also like to thank the GeoHab Organizing Committee for providing funding for A. Ilich to attend the GeoHab 2022 conference in Venice, Italy, via the Ron McDowell Award which provides funding for students to attend GeoHab, giving students the opportunity to present, learn, and receive valuable feedback from experts in their field. Lastly, we would like to thank Dr. Jakub Nowosad for his assistance with submitting the package to CRAN. This research was supported by the National Fish and Wildlife Foundation through the Gulf Environmental Benefit Fund (Grant GEBF45892 to S. Murawski, C. Lembke, and S. Locker, “Restoring Fish and Sea Turtle Habitat on the West Florida Continental Shelf: Benthic Habitat Mapping, Characterization and Assessment”) as well as by a cooperative agreement between NOAA's Office of Coast Survey and the University of South Florida through the Center for Ocean Mapping and Innovative Technologies (COMIT), NA20NOS4000227. A. Ilich was also supported by the William and Elsie Knight Endowed Fellowship Fund for Marine Science, Young Fellowship Program Fund, and the Von Rosenstiel Endowed Fellowship provided by the College of Marine Science at the University of South Florida.

CONFLICT OF INTEREST STATEMENT

The authors have no conflict of interest to declare.

DATA AVAILABILITY STATEMENT

All data and code to run the analyses contained within this article have been included in the supplementary material. Source code for the `MultiscaleDTM` R package is available both on github (https://github.com/ailich/MultiscaleDTM) and on Zenodo (https://doi.org/10.5281/zenodo.5548338).

ORCID

Alexander R. Ilich https://orcid.org/0000-0003-1758-8499
Benjamin Misiuk https://orcid.org/0000-0001-5822-4574
Vincent Lecours https://orcid.org/0000-0002-4777-3348
Steven A. Murawski https://orcid.org/0000-0001-7002-4356

REFERENCES

Albani, M., Klinkenberg, B., Andison, D. W., & Kimmins, J. P. (2004). The choice of window size in approximating topographic surfaces from digital elevation models. *International Journal of Geographical Information Science*, *18*, 577–593. https://doi.org/10.1080/13658810410001701987

Behrens, T., Schmidt, K., MacMillan, R. A., & Viscarra Rossel, R. A. (2018). Multiscale contextual spatial modelling with the gaussian scale space. *Geoderma*, *310*, 128–137. https://doi.org/10.1016/j.geoderma.2017.09.015

Behrens, T., Viscarra Rossel, R. A., Kerry, R., MacMillan, R., Schmidt, K., Lee, J., Scholten, T., & Zhu, A.-X. (2019). The relevant range of scales for multi-scale contextual spatial modelling. *Scientific Reports*, *9*, 14800. https://doi.org/10.1038/s41598-019-51395-3

Behrens, T., Zhu, A.-X., Schmidt, K., & Scholten, T. (2010). Multi-scale digital terrain analysis and feature selection for digital soil mapping. *Geoderma*, *155*, 175–185. https://doi.org/10.1016/j.geoderma.2009.07.010

Bishop, M. P., James, L. A., Shroder, J. F., Jr., & Walsh, S. J. (2012). Geospatial technologies and digital geomorphological mapping: Concepts, issues and research. *Geomorphology*, *137*, 5–26. https://doi.org/10.1016/j.geomorph.2011.06.027

Bivand, R. S. (2021). Progress in the R ecosystem for representing and handling spatial data. *Journal of Geographical Systems*, *23*, 515–546. https://doi.org/10.1007/s10109-020-00336-0

Bouchet, P. J., Meeuwig, J. J., Salgado Kent, C. P., Letessier, T. B., & Jenner, C. K. (2015). Topographic determinants of mobile vertebrate predator hotspots: Current knowledge and future directions. *Biological Reviews*, *90*, 699–728. https://doi.org/10.1111/brv.12130

Brown, C. J., Smith, S. J., Lawton, P., & Anderson, J. T. (2011). Benthic habitat mapping: A review of progress towards improved understanding of the spatial ecology of the seafloor using acoustic techniques. *Estuarine, Coastal and Shelf Science*, *92*, 502–520. https://doi.org/10.1016/j.ecss.2011.02.007

Cavalli, M., Tarolli, P., Marchi, L., & Dalla Fontana, G. (2008). The effectiveness of airborne LiDAR data in the recognition of channel-bed morphology. *Catena*, *73*, 249–260. https://doi.org/10.1016/j.catena.2007.11.001

Cavazzi, S., Corstanje, R., Mayr, T., Hannam, J., & Fealy, R. (2013). Are fine resolution digital elevation models always the best choice in digital soil mapping? *Geoderma*, *195–196*, 111–121. https://doi.org/10.1016/j.geoderma.2012.11.020

Conrad, O., Bechtel, B., Bock, M., Dietrich, H., Fischer, E., Gerlitz, L., Wehberg, J., Wichmann, V., & Böhner, J. (2015). System for automated geoscientific analyses (SAGA) v. 2.1.4. *Geoscientific Model Development*, *8*, 1991–2007. https://doi.org/10.5194/gmd-8-1991-2015

Csillag, F., & Kabos, S. (2002). Wavelets, boundaries, and the spatial analysis of landscape pattern. *Écoscience*, *9*, 177–190. https://doi.org/10.1080/11956860.2002.11682704

Dolan, M. F. J. (2012). *Calculation of slope angle from bathymetry data using GIS—Effects of computation algorithms, data resolution and analysis scale* (NGU Report No. 2012.041). Geological Survey of Norway.

Dove, D., Nanson, R., Bjarnadóttir, L. R., Guinan, J., Gafeira, J., Post, A., Dolan, M. F. J., Stewart, H., Arosio, R., & Scott, G. (2020). *A two-part seabed geomorphology classification scheme (v.2); part 1: Morphology features glossary (open report)*. MIM-GA Geomorphology Group. https://doi.org/10.5281/ZENODO.4075248

Drăguţ, L., Eisank, C., & Strasser, T. (2011). Local variance for multi-scale analysis in geomorphometry. *Geomorphology*, *130*, 162–172. https://doi.org/10.1016/j.geomorph.2011.03.011

Du Preez, C. (2015). A new arc–chord ratio (ACR) rugosity index for quantifying three-dimensional landscape structural complexity. *Landscape Ecology*, *30*, 181–192. https://doi.org/10.1007/s10980-014-0118-8

Eddelbuettel, D., François, R., Allaire, J., Ushey, K., Kou, Q., Russel, N., Chambers, J., & Bates, D. (2011). Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, *40*, 1–18. https://doi.org/10.18637/jss.v040.i08

Eddelbuettel, D., & Sanderson, C. (2014). RcppArmadillo: Accelerating R with high-performance C++ linear algebra. *Computational Statistics & Data Analysis*, *71*, 1054–1063. https://doi.org/10.1016/j.csda.2013.02.005

Evans, I. S. (1980). An integrated system of terrain analysis and slope mapping. *Zeitschrift für Geomorphologie. Supplementband Stuttgart*, *36*, 274–295.

Federal Geographic Data Committee. (2012). *Coastal and marine ecological classification standard* (No. FGDC-STD-018-2012).

Fisher, P., Wood, J., & Cheng, T. (2005). Fuzziness and ambiguity in multi-scale analysis of landscape morphometry. In F. E. Petry, V. B. Robinson, & M. A. Cobb (Eds.), *Fuzzy modeling with spatial information for geographic problems* (pp. 209–232). Springer. https://doi.org/10.1007/3-540-26886-3_10

Fleming, M. D., & Hoffer, R. M. (1979). *Machine processing of landsat MSS data and DMA topographic data for forest cover type mapping* (LARS Technical Report 062879). Laboratory for Applications of Remote Sensing, Purdue University.

Florinsky, I. V. (2017). Spheroidal equal angular DEMs: The specificity of morphometric treatment. *Transactions in GIS*, *21*, 1115–1129. https://doi.org/10.1111/tgis.12269

Franklin, J. (1995). Predictive vegetation mapping: Geographic modelling of biospatial patterns in relation to environmental gradients. *Progress in Physical Geography: Earth and Environment*, *19*, 474–499. https://doi.org/10.1177/030913339501900403

Friedman, A., Pizarro, O., Williams, S. B., & Johnson-Roberson, M. (2012). Multi-scale measures of rugosity, slope and aspect from benthic stereo image reconstructions. *PLoS ONE*, *7*, e50440. https://doi.org/10.1371/journal.pone.0050440

GDAL/OGR Contributors. (2000). *GDAL/OGR geospatial data abstraction software library*. Open Source Geospatial Foundation. https://doi.org/10.5281/zenodo.5884351

Gonçalves, J., Pôças, I., Marcos, B., Mücher, C. A., & Honrado, J. P. (2019). SegOptim—A new R package for optimizing object-based image analyses of high-spatial resolution remotely-sensed data. *International Journal of Applied Earth Observation and Geoinformation*, *76*, 218–230. https://doi.org/10.1016/j.jag.2018.11.011

Goodchild, M. F. (2011). Scale in GIS: An overview. *Geomorphology, Scale Issues in Geomorphology*, *130*, 5–9. https://doi.org/10.1016/j.geomorph.2010.10.004

Grohmann, C. H. (2004). Morphometric analysis in geographic information systems: Applications of free software GRASS and R. *Computers & Geosciences*, *30*, 1055–1067. https://doi.org/10.1016/j.cageo.2004.08.002

Grohmann, C. H., & Riccomini, C. (2009). Comparison of roving-window and search-window techniques for characterising landscape morphometry. *Computers & Geosciences*, *35*, 2164–2169. https://doi.org/10.1016/j.cageo.2008.12.014

Grohmann, C. H., Smith, M. J., & Riccomini, C. (2011). Multiscale analysis of topographic surface roughness in the Midland Valley, Scotland. *IEEE Transactions on Geoscience and Remote Sensing*, *49*, 1200–1213. https://doi.org/10.1109/TGRS.2010.2053546

Guth, P. (2009). Geomorphometry in MicroDEM. In T. Hengl & H. I. Reuter (Eds.), *Developments in soil science, geomorphometry* (pp. 351–366). Elsevier. https://doi.org/10.1016/S0166-2481(08)00015-9

Guth, P., & Kane, M. (2021). Slope, aspect, and hillshade algorithms for non-square digital elevation models. *Transactions in GIS*, *25*, 2309–2332. https://doi.org/10.1111/tgis.12852

Habib, M. (2021). Quantifying topographic ruggedness using principal component analysis. *Advances in Civil Engineering*, *2021*, e3311912. https://doi.org/10.1155/2021/3311912

Halley, J. M., Hartley, S., Kallimanis, A. S., Kunin, W. E., Lennon, J. J., & Sgardelis, S. P. (2004). Uses and abuses of fractal methodology in ecology. *Ecology Letters*, *7*, 254–271. https://doi.org/10.1111/j.1461-0248.2004.00568.x

Hengl, T. (2011). *A practical guide to geostatistical mapping* (2nd ed.). Lulu.

Hijmans, R. J. (2010). *Raster: Geographic data analysis and modeling*. https://rspatial.org/raster

Hijmans, R. J. (2020). *Terra: Spatial data analysis*. https://rspatial.org/terra/

Hobson, R. D. (1972). Surface roughness in topography: Quantitative approach. In R. J. Chorley (Ed.), *Spatial analysis in geomorphology* (p. 404). Routledge.

Horn, B. K. (1981). Hill shading and the reflectance map. *IEEE*, *69*, 14–47. https://doi.org/10.1109/PROC.1981.11918

Ilich, A. R., Misiuk, B., Lecours, V., & Murawski, S. A. (2021). *MultiscaleDTM*. https://doi.org/10.5281/zenodo.5548338

Ironside, K. E., Mattson, D. J., Arundel, T., Theimer, T., Holton, B., Peters, M., Edwards, T. C., & Hansen, J. (2018). Geomorphometry in landscape ecology: Issues of scale, physiography, and application. *Environment and Ecology Research*, *6*, 397–412. https://doi.org/10.13189/eer.2018.060501

Jasiewicz, J., & Stepinski, T. F. (2013). Geomorphons—A pattern recognition approach to classification and mapping of landforms. *Geomorphology*, *182*, 147–156. https://doi.org/10.1016/j.geomorph.2012.11.005

Jenness, J. S. (2004). Calculating landscape surface area from digital elevation models. *Wildlife Society Bulletin*, *32*, 829–839. https://doi.org/10.2193/0091-7648(2004)032%5B0829:CLSAFD%5D2.0.CO;2

Karney, C. F. F. (2013). Algorithms for geodesics. *Journal of Geodesy*, *87*, 43–55. https://doi.org/10.1007/s00190-012-0578-z

Katzil, Y., & Doytsher, Y. (2000). Height estimation methods for filling gaps in gridded DTM. *Journal of Surveying Engineering*, *126*, 145–162. https://doi.org/10.1061/(ASCE)0733-9453(2000)126:4(145)

Kovesi, P. (2009). *Arbitrary gaussian filtering with 25 additions and 5 multiplications per pixel*. School of Computer Science and Software Engineering at the University of Western Australia.

Krcho, J. (1983). Teoreticka koncepcia a interdisciplinarne aplikacie komplexneho digitalneho modelu reliefu pri modelovani dvojdimenzionalnych poli. *Geograficky casopis*, *35*, 265–291.

Krcho, J. (1999). Modelling of georelief using DTM—The influence of point configuration of input points field on positional and numeric accuracy. *Geografický časopis*, *51*, 225–260.

Kuhn, M. (2008). Building predictive models in R using the caret package. *Journal of Statistical Software*, *28*, 1–26. https://doi.org/10.18637/jss.v028.i05

Kuhn, M., & Vaughan, D. (2022). *Parsnip: A common API to modeling and analysis functions*. https://CRAN.R-project.org/package=parsnip

Lai, J., Lortie, C. J., Muenchen, R. A., Yang, J., & Ma, K. (2019). Evaluating the popularity of R in ecology. *Ecosphere*, *10*, e02567. https://doi.org/10.1002/ecs2.2567

Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., Au, Q., Casalicchio, G., Kotthoff, L., & Bischl, B. (2019). Mlr3: A modern object-oriented machine learning framework in R. *Journal of Open Source Software*, *4*, 1903. https://doi.org/10.21105/joss.01903

Lecours, V., Devillers, R., Schneider, D. C., Lucieer, V. L., Brown, C. J., & Edinger, E. N. (2015). Spatial scale and geographic context in benthic habitat mapping: Review and future directions. *Marine Ecology Progress Series*, *535*, 259–284. https://doi.org/10.3354/meps11378

Lecours, V., Devillers, R., Simms, A. E., Lucieer, V. L., & Brown, C. J. (2017). Towards a framework for terrain attribute selection in environmental studies. *Environmental Modelling & Software*, *89*, 19–30. https://doi.org/10.1016/j.envsoft.2016.11.027

Lecours, V., Dolan, M. F., Micallef, A., & Lucieer, V. L. (2016). A review of marine geomorphometry, the quantitative study of the seafloor. *Hydrology and Earth System Sciences*, *20*, 3207–3244. https://doi.org/10.5194/hess-20-3207-2016

Leutner, B., & Horning, N. (2016). *RStoolbox: Tools for remote sensing data analysis*. https://bleutner.github.io/RStoolbox/

Levin, S. A. (1992). The problem of pattern and scale in ecology. *Ecology*, *73*, 1943–1967. https://doi.org/10.2307/1941447

Lindsay, J. B. (2016). Whitebox GAT: A case study in geomorphometric analysis. *Computers & Geosciences*, *95*, 75–84. https://doi.org/10.1016/j.cageo.2016.07.003

Lindsay, J. B., Cockburn, J. M. H., & Russell, H. A. J. (2015). An integral image approach to performing multi-scale topographic position analysis. *Geomorphology*, *245*, 51–61. https://doi.org/10.1016/j.geomorph.2015.05.025

Lindsay, J. B., Newman, D. R., & Francioni, A. (2019). Scale-optimized surface roughness for topographic analysis. *Geosciences*, *9*, 322. https://doi.org/10.3390/geosciences9070322

Lovelace, R., Nowosad, J., & Muenchow, J. (2019). *Geocomputation with R*. CRC Press. https://doi.org/10.1201/9780203730058

Lundblad, E. R., Wright, D. J., Miller, J., Larkin, E. M., Rinehart, R., Naar, D. F., Donahue, B. T., Anderson, S. M., & Battista, T. (2006). A benthic terrain classification scheme for American Samoa. *Marine Geodesy*, *29*, 89–111. https://doi.org/10.1080/01490410600738021

Mandelbrot, B. (1967). How long is the coast of Britain? Statistical self-similarity and fractional dimension. *Science*, *156*, 636–638. https://doi.org/10.1126/science.156.3775.636

Masetti, G., Mayer, L., & Ward, L. (2018). A bathymetry-and reflectivity-based approach for seafloor segmentation. *Geosciences*, *8*, 14. https://doi.org/10.3390/geosciences8010014

Maxwell, A. E., & Shobe, C. M. (2022). Land-surface parameters for spatial predictive mapping and modeling. *Earth-Science Reviews*, *226*, 103944. https://doi.org/10.1016/j.earscirev.2022.103944

McKean, J., & Roering, J. (2004). Objective landslide detection and surface morphology mapping using high-resolution airborne laser altimetry. *Geomorphology*, *57*, 331–351. https://doi.org/10.1016/S0169-555X(03)00164-8

Minár, J., Evans, I. S., & Jenčo, M. (2020). A comprehensive system of definitions of land surface (topographic) curvatures, with implications for their application in geoscience modelling and prediction. *Earth-Science Reviews*, *211*, 103414. https://doi.org/10.1016/j.earscirev.2020.103414

Misiuk, B., Lecours, V., & Bell, T. (2018). A multiscale approach to mapping seabed sediments. *PLoS ONE*, *13*, e0193647. https://doi.org/10.1371/journal.pone.0193647

Misiuk, B., Lecours, V., Dolan, M. F. J., & Robert, K. (2021). Evaluating the suitability of multi-scale terrain attribute calculation approaches for seabed mapping applications. *Marine Geodesy*, *44*, 327–385. https://doi.org/10.1080/01490419.2021.1925789

Mod, H. K., Scherrer, D., Luoto, M., & Guisan, A. (2016). What we use is not what we know: Environmental predictors in plant distribution models. *Journal of Vegetation Science*, *27*, 1308–1322. https://doi.org/10.1111/jvs.12444

Moudrý, V., Lecours, V., Malavasi, M., Misiuk, B., Gábor, L., Gdulová, K., Šímová, P., & Wild, J. (2019). Potential pitfalls in rescaling digital terrain model-derived attributes for ecological studies. *Ecological Informatics*, *54*, 100987. https://doi.org/10.1016/j.ecoinf.2019.100987

Neteler, M., Bowman, M. H., Landa, M., & Metz, M. (2012). GRASS GIS: A multi-purpose open source GIS. *Environmental Modelling & Software*, *31*, 124–130. https://doi.org/10.1016/j.envsoft.2011.11.014

Newman, D. R., Cockburn, J. M. H., Drăguţ, L., & Lindsay, J. B. (2022). Evaluating scaling frameworks for multiscale geomorphometric analysis. *Geomatics*, *2*, 36–51. https://doi.org/10.3390/geomatics2010003

NOAA NCEI. (2014). *Updated Digital Elevation Model (CUDEM)—1/9 Arc-Second Resolution Bathymetric-Topographic Tiles* [northeast_sandy]. https://doi.org/10.25921/ds9v-ky35

Nowosad, J., & Stepinski, T. (2021). *Generalizing the simple linear iterative clustering (SLIC) superpixels*. UC Santa Barbara: Center for Spatial Studies. https://doi.org/10.25436/E2QP4R

Olaya, V. (2009). Basic land-surface parameters. In T. Hengl & H. I. Reuter (Eds.), *Geomorphometry: Concepts, software, applications, developments in soil science*. Elsevier. https://doi.org/10.1016/S0166-2481(08)00006-8

Pike, R. J., Evans, I. S., & Hengl, T. (2009). Chapter 1 Geomorphometry: A brief guide. In T. Hengl & H. I. Reuter (Eds.), *Developments in soil science* (pp. 3–30). Elsevier. https://doi.org/10.1016/S0166-2481(08)00001-9

QGIS Development Team. (2009). *QGIS geographic information system*. Open Source Geospatial Foundation. http://qgis.org

R Core Team. (2022). *R: A language and environment for statistical computing*. https://www.R-project.org/

Risk, M. J. (1972). Fish diversity on a coral reef in the Virgin Islands. *Atoll Research Bulletin*, *153*, 1–4. https://doi.org/10.5479/si.00775630.153.1

Ritter, P. (1987). A vector-based slope and aspect generation algorithm. *Photogrammetric Engineering and Remote Sensing*, *53*, 1109–1111.

Sanderson, C., & Curtin, R. (2016). Armadillo: A template-based C++ library for linear algebra. *The Journal of Open Source Software*, *1*, 26. https://doi.org/10.21105/joss.00026

Sanderson, C., & Curtin, R. (2020). An adaptive solver for systems of linear equations. *14th International Conference on Signal Processing and Communication Systems*, Adelaide, SA, Australia, (pp. 1–6). https://doi.org/10.1109/ICSPCS50536.2020.9309998

Sappington, J. M., Longshore, K. M., & Thompson, D. B. (2007). Quantifying landscape ruggedness for animal habitat analysis: A case study using Bighorn sheep in the Mojave Desert. *The Journal of Wildlife Management*, *71*, 1419–1426. https://doi.org/10.2193/2005-723

Schmidt, J., & Andrew, R. (2005). Multi-scale landform characterization. *Area*, *37*, 341–350. https://doi.org/10.1111/j.1475-4762.2005.00638.x

Shang, X., Robert, K., Misiuk, B., Mackin-McLaughlin, J., & Zhao, J. (2021). Self-adaptive analysis scale determination for terrain features in seafloor substrate classification. *Estuarine, Coastal and Shelf Science*, *254*, 107359. https://doi.org/10.1016/j.ecss.2021.107359

Shary, P. A. (1995). Land surface in gravity points classification by a complete system of curvatures. *Mathematical Geology*, *27*, 373–390. https://doi.org/10.1007/BF02084608

Sîrbu, F., Drăguţ, L., Oguchi, T., Hayakawa, Y., & Micu, M. (2019). Scaling land-surface variables for landslide detection. *Progress in Earth and Planetary Science*, *6*, 44. https://doi.org/10.1186/s40645-019-0290-1

Smith, M. W. (2014). Roughness in the earth sciences. *Earth-Science Reviews*, *136*, 202–225. https://doi.org/10.1016/j.earscirev.2014.05.016

Sofia, G. (2020). Combining geomorphometry, feature extraction techniques and earth-surface processes research: The way forward. *Geomorphology*, *355*, 107055. https://doi.org/10.1016/j.geomorph.2020.107055

Stroustrup, B. (2013). *The C++ programming language* (4th ed.). Addison-Wesley.

Tippmann, S. (2015). Programming tools: Adventures with R. *Nature*, *517*, 109–110. https://doi.org/10.1038/517109a

Trevisani, S., Teza, G., & Guth, P. (2022). *A simplified geostatistical approach for characterizing key aspects of short-range roughness*. https://doi.org/10.2139/ssrn.4223135

Vincenty, T. (1975). Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, *23*, 88–93. https://doi.org/10.1179/sre.1975.23.176.88

Walbridge, S., Slocum, N., Pobuda, M., & Wright, D. J. (2018). Unified geomorphological analysis workflows with benthic terrain modeler. *Geosciences*, *8*, 94. https://doi.org/10.3390/geosciences8030094

Wang, D., Laffan, S. W., Liu, Y., & Wu, L. (2010). Morphometric characterisation of landform from DEMs. *International Journal of Geographical Information Science*, *24*, 305–326. https://doi.org/10.1080/13658810802467969

Weiss, A. (2001). Topographic position and landforms analysis. *ESRI User Conference*, San Diego, CA.

Wiens, J. A. (1989). Spatial scaling in ecology. *Functional Ecology*, *3*, 385–397. https://doi.org/10.2307/2389612

Wilson, J. P. (2018). *Environmental applications of digital terrain modeling, First edition, new analytical methods in earth and environmental science*. John Wiley & Sons Ltd. https://doi.org/10.1002/9781118938188

Wilson, J. P., & Gallant, J. C. (Eds.). (2000). *Terrain analysis: Principles and applications*. John Wiley & Sons, Inc.

Wilson, M. F., O'Connell, B., Brown, C., Guinan, J. C., & Grehan, A. J. (2007). Multiscale terrain analysis of multibeam bathymetry data for habitat mapping on the continental slope. *Marine Geodesy*, *30*, 3–35. https://doi.org/10.1080/01490410701295962

Wood, J. (1996). *The geomorphological characterisation of digital elevation models* [PhD thesis]. University of Leicester.

Wood, J. (2009). Chapter 14 Geomorphometry in LandSerf. In T. Hengl & H. I. Reuter (Eds.), *Developments in soil science, geomorphometry* (pp. 333–349). Elsevier. https://doi.org/10.1016/S0166-2481(08)00014-7

Zhang, J., Ye, Z., & Zheng, K. (2021). A parallel computing approach to spatial neighboring analysis of large amounts of terrain data using spark. *Sensors (Basel, Switzerland)*, *21*, 365. https://doi.org/10.3390/s21020365

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Ilich, A. R., Misiuk, B., Lecours, V., & Murawski, S. A. (2023). MultiscaleDTM: An open-source R package for multiscale geomorphometric analysis. *Transactions in GIS*, *27*, 1164–1204. https://doi.org/10.1111/tgis.13067

APPENDIX A

SlpAsp

$$dz.dx = \frac{\sum \left(r^* w_{\mathrm{right}}\right) - \sum \left(r^* w_{\mathrm{left}}\right)}{\left(\sum \left(w_{\mathrm{right}} + w_{\mathrm{left}}\right)\right) * x_{\mathrm{res}}{}^* \frac{n_{\mathrm{col}} - 1}{2}}$$

$$dz.dy = \frac{\sum \left(r^* w_{\mathrm{top}}\right) - \sum \left(r^* w_{\mathrm{bottom}}\right)}{\left(\sum \left(w_{\mathrm{top}} + w_{\mathrm{bottom}}\right)\right) * y_{\mathrm{res}}{}^* \frac{n_{\mathrm{row}} - 1}{2}}$$

where r is the focal window, w is the weights matrices, n_{row} is the number of rows in the focal window, n_{col} is the number of columns in the focal window, x_{res} is the x cell resolution and y_{res} is the y cell resolution.

$$r = \begin{bmatrix} c_1 & c_2 & c_3 \\ c_4 & c_5 & c_6 \\ c_7 & c_8 & c_9 \end{bmatrix}$$

Note: Values c_1–c_9 correspond to the values used in the focal window shown in Figure 3 in the main text.

$$w_{\mathrm{left}} = \begin{cases} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \text{if method is rook} \\ \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} & \text{if method is queen} \end{cases}$$

$$w_{\text{right}} = \begin{cases} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} & \text{if method is rook} \\ \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} & \text{if method is queen} \end{cases}$$

$$w_{\text{top}} = \begin{cases} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \text{if method is rook} \\ \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \text{if method is queen} \end{cases}$$

$$w_{\text{bottom}} = \begin{cases} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} & \text{if method is rook} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} & \text{if method is queen} \end{cases}$$

$$\text{Slope} = \text{atan}\left(\sqrt{dz.dx^2 + dz.dy^2}\right)$$

$$\text{Aspect} = \frac{-\pi}{2} - \text{atan2}(dz.dy, dz.dx)$$

Note: If aspect < 0 add 2π.

Qfit

$$\text{Slope} = \arctan\left(\sqrt{d^2 + e^2}\right)$$

$$\text{Aspect} = \frac{-\pi}{2} - \text{atan2}(e, d)$$

Note: If aspect < 0 add 2π.

$$\text{Profile curvature} = \frac{-2\left(ad^2 + cde + be^2\right)}{\left(d^2 + e^2\right)\left(1 + d^2 + e^2\right)^{\frac{3}{2}}}$$

$$\text{Planform curvature} = \frac{-2\left(ae^2 - cde + bd^2\right)}{\left(d^2+e^2\right)\sqrt{1+d^2+e^2}}$$

$$\text{Twisting curvature} = \frac{2de(a-b) - c\left(d^2-e^2\right)}{\left(d^2+e^2\right)\left(1+d^2+e^2\right)}$$

$$\text{Mean curvature} = -\frac{a\left(1+e^2\right) - cde + b\left(1+d^2\right)}{\left(1+d^2+e^2\right)^{\frac{3}{2}}}$$

$$\text{ku} = \sqrt{\left(\frac{a\left(1+e^2\right) - cde + b\left(1+d^2\right)}{\left(1+d^2+e^2\right)^{\frac{3}{2}}}\right)^2 - \frac{4ab - c^2}{\left(1+d^2+e^2\right)^2}}$$

$$\text{Max curvature} = \text{Mean curvature} + \text{ku}$$

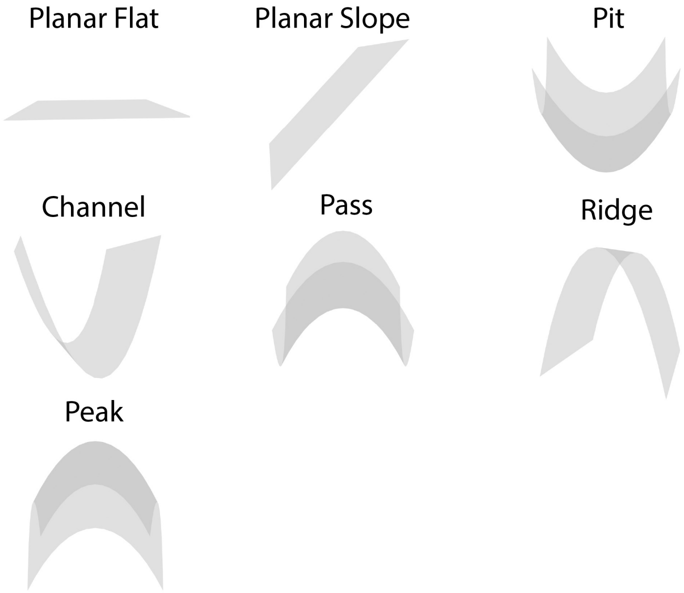
$$\text{Min curvature} = \text{Mean curvature} - \text{ku}$$

APPENDIX B

Morphometric features

Morphometric features based on the slope and several types of curvatures (planform, maximum, minimum) and a user specified tolerance threshold for slope and curvature that define a flat/planar surface (Wood, 1996). These classes are planar flat, planar slope, pit, channel, pass, ridge, and peak. The morphometric features algorithm has been modified from Wood (1996) to use the curvature measures defined previously, and the "planar" class is additionally split into "planar flat" and "planar slope" as slope is considered its own class in several similar landform and habitat mapping schemes (Dove et al., 2020; Federal Geographic Data Committee, 2012; Jasiewicz & Stepinski, 2013; Masetti et al., 2018) and is an important factor shaping biotic communities (Moudrý et al., 2019; Sappington et al., 2007; Wilson et al., 2007).

A visual representation of the seven types of morphometric features is shown below.



The mathematical definitions in terms of slope, planform, maximum, and minimum curvatures are provided in the table below. In order to classify these morphometric features, a user set threshold for both slope and curvature for what defines a flat/planar surface must be defined. In the table, "+" indicates greater than the threshold, "–" indicates less than the −1 * theshold, and "0" indicates between −1 * threshold and threshold.

Feature	Slope	Plan curvature	Max. curvature	Min. curvature
Planar flat	0		0	0
Planar slope	+	0		
Pit	0		–	–
Channel	+	–		
	0		0	–
Pass	0		+	–
Ridge	+	+		
	0		+	0
Peak	0		+	+

APPENDIX C

Advanced use cases

```
library(dplyr)

library(terra)

library(MultiscaleDTM)



DTM<- rast("Data/fishersisland_dtm.tif")
```

Custom measures

The results of the local quadratic fit can be expressed either as regression parameters a–e in the equation $Z = aX^2 + bY^2 + cXY + dx + eY + \text{intercept}$ or via partial derivatives with respect to x^2, y^2, xy, x, and y. Converting between these two representations is straightforward at the focal cell as the local coordinate system defines the focal cell as the origin ($x = 0$, $y = 0$). Therefore, this reduces to $z_{xx} = 2a$, $z_{yy} = 2b$, $z_{xy} = c$, $z_x = d$, and $z_y = e$ where z represents the partial derivative with respect to the subscript.

Custom functions can be defined to calculate alternative terrain attributes based on the local first and second order partial derivatives of the surface. For example, Cross-sectional curvature and Total Gaussian curvature can be calculated based on their respective equations using regression parameters a–e from Wood (1996) or from the partial derivatives based on the equations of Minár et al. (2020).

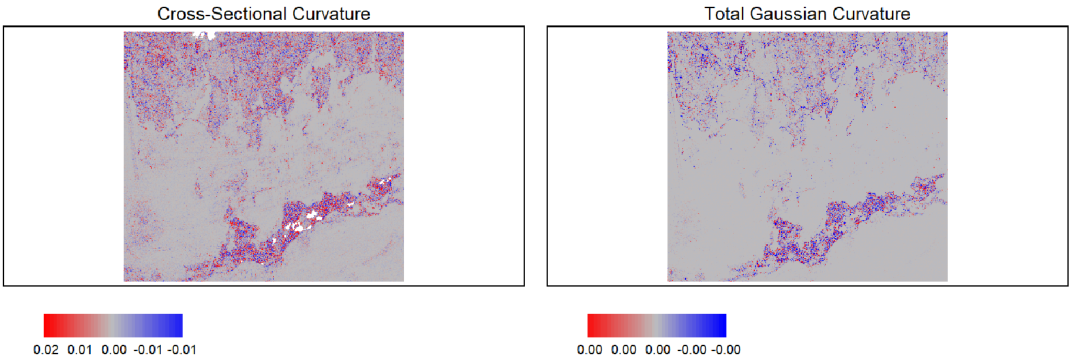
```
CrossCurv<- function(a,b,c,d,e){

crosc<- -2*(b*d^2+a*e^2-c*d*e)/(d^2+e^2)

return(crosc)

} #Cross-sectional curvature function based on formula


TotalCurv<- function(zxx, zyy, zxy, zx, zy){

tc<- (zxx*zyy-zxy^2)/(1+zx^2+zy^2)^2

return(tc)

} #Define total Gaussian curvature function based on formula
```

We can calculate the regression parameters or the partial derivatives using the `Qfit` function.

```
#Calculate regression parameters a-e

reg_params<- Qfit(DTM, metrics= c(), return_params = TRUE)

#Calculate partial derivatives

partial_derivs<- Qfit(DTM, metrics= c(), return_params = TRUE,

                     as_derivs = TRUE)
```

The custom functions can then be applied across these layers using `lapp` (layer apply) function from the `terra` package.

```
#Cross-sectional curvature

crosc<- lapp(reg_params, CrossCurv)


#Total Gaussian curvature

tc<- lapp(partial_derivs, TotalCurv)
```



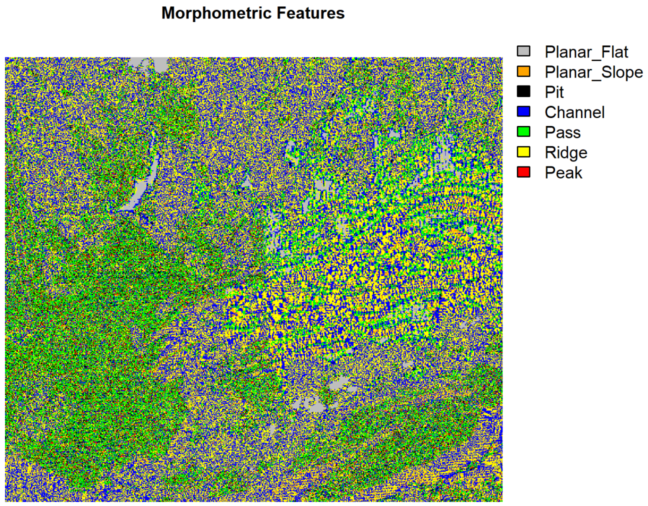
Classification dictionary

Classification dictionaries can be defined using conditional statements that assign each cell a class based on the values of geomorphometric (or other) variables. In fact this is how morphometric features are defined internally within the `Qfit` function. We could do this manually by calculating the slope, planform, maximum, and minimum curvature and then creating a function with the appropriate conditional statements to define the seven classes from Appendix B.

```
#Calculate select terrain attributes

TA<- Qfit(DTM, metrics = c("qslope", "planc", "maxc", "minc"))


#Set up classification dictionary based on conditional statements

#condition on left and class on right of "~"


classify_features<- function(slope, planc, maxc, minc){

#Cell values and their corresponding class

#1=PLANAR_FLAT

#2=PLANAR_SLOPE

#3=PIT

#4=CHANNEL

#5=PASS

#6=RIDGE

#7=PEAK

case_when(is.na(slope) ~ NA_real_,

          (slope > 1) & (planc > 1e-04) ~ 6,

          (slope > 1) & (planc < -1e-04) ~ 4,

          slope > 1 ~ 2,

          (maxc > 1e-04) & (minc > 1e-04) ~ 7,

          (maxc > 1e-04) & (minc < -1e-04) ~ 5,

          maxc > 1e-04 ~ 6,

          (minc < -1e-04) & (maxc < -1e-04) ~ 3,

          minc < -1e-04 ~ 4,

          TRUE ~ 1)

}
```

```
#Apply function across layers

features<- lapp(TA, fun = classify_features)


#Define factor levels

levels(features)<- data.frame(ID=1:7,

                              features= c("Planar_Flat",

                                          "Planar_Slope", "Pit",

                                          "Channel", "Pass",

                                          "Ridge","Peak"))
```

We can then use the `lapp` function to apply this function across the layers and define the factor levels so that number values have an associated class.


Morphometric Features
Planar_Flat
Planar_Slope
Pit
Channel
Pass
Ridge
Peak