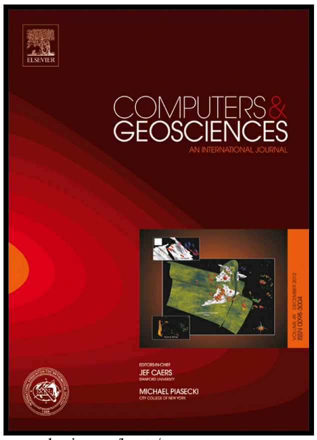
Author's Accepted Manuscript



Spatial ecological complexity measures in GRASS GIS

Duccio Rocchini, Vaclav Petras, Anna Petrasova, Yann Chemin, Carlo Ricotta, Alessandro Frigeri, Martin Landa, Matteo Marcantonio, Lucy Bastin, Markus Metz, Luca Delucchi, Markus Neteler

PII: S0098-3004(16)30128-5
DOI: http://dx.doi.org/10.1016/j.cageo.2016.05.006
Reference: CAGEO3761

To appear in: *Computers and Geosciences*

Received date: 12 November 2015
Revised date: 15 April 2016
Accepted date: 13 May 2016

Cite this article as: Duccio Rocchini, Vaclav Petras, Anna Petrasova, Yann Chemin, Carlo Ricotta, Alessandro Frigeri, Martin Landa, Matteo Marcantonio Lucy Bastin, Markus Metz, Luca Delucchi and Markus Neteler, Spatia ecological complexity measures in GRASS GIS, *Computers and Geosciences* http://dx.doi.org/10.1016/j.cageo.2016.05.006

Spatial ecological complexity measures in GRASS GIS

Duccio Rocchini[1,*], Vaclav Petras[2,3], Anna Petrasova[2,3], Yann Chemin[4], Carlo Ricotta[5], Alessandro Frigeri[6], Martin Landa[7], Matteo Marcantonio[1], Lucy Bastin[8], Markus Metz[1], Luca Delucchi[1], and Markus Neteler[1]

[1]Fondazione Edmund Mach, Department of Biodiversity and Molecular Ecology, Research and Innovation Centre, Via E. Mach 1, 38010 S. Michele allâĂŽAdige (TN), Italy, corresponding author: ducciorocchini@gmail.com, duccio.rocchini@fmach.it
[2]Department of Marine, Earth, and Atmospheric Sciences, North Carolina State University, 2800 Faucette Drive, Raleigh, NC 27695, USA
[3]Center for Geospatial Analytics, North Carolina State University, 2800 Faucette Drive, Raleigh, NC 27695, USA
[4]IWMI, Research Division, Pelawatta, Sri Lanka
[5]Department of Environmental Biology, University of Rome "La Sapienza", 00185 Rome, Italy
[6]Istituto Nazionale di Astrofisica (INAF), Istituto di Astrofisica e Planetologia Spaziali (IAPS), Via Fosso del Cavaliere 100, 00133 Rome, Italy
[7]Department of Mapping and Cartography and Faculty of Civil Engineering, Czech Technical University in Prague, 166 29 Prague, Czech Republic
[8]School of Computer Science, Aston University, UK, (currently on secondment to Joint Research Centre of the European Commission)
[*]corresponding author: ducciorocchini@gmail.com, duccio.rocchini@fmach.it

May 18, 2016

Abstract

Good estimates of ecosystem complexity are essential for a number of ecological tasks: from biodiversity estimation, to forest structure variable retrieval, to feature extraction by edge detection and generation of multifractal surface as neutral models for e.g. feature change assessment. Hence, measuring ecological complexity over space becomes crucial in macroecology and geography. Many geospatial tools have been advocated in spatial ecology to estimate ecosystem complexity and its changes over space and time. Among these tools, free and open source options especially offer opportunities to guarantee the robustness of algorithms and reproducibility. In this paper we will summarize the most straightforward measures of spatial complexity available in the Free and Open Source Software GRASS GIS, relating them to key ecological patterns and processes.

keywords: Free and Open Source Software; remote sensing; spatial complexity; spatial ecology.

1 Introduction

In spatial ecology, the complexity of ecosystems, and the changes in that complexity over time, are critical issues. Mapping and modelling landscape heterogeneity over space and time has been acknowledged as one of the most powerful methods to gather information about underlying changes in abiotic and biotic components of ecosystems including land cover, land use, vegetation and soil.

Experimental manipulations to effectively measure complexity in the field are difficult from both a cost and a logistical point of view, and, depending on the scale of the studied ecological problem, may become impossible (Rocchini et al., 2013a). Therefore, proxies for ecological complexity are needed. Reliable proxy variables which are available at large scale can allow upscaling of complexity estimates and a clearer focus on processes that act at multiple spatial and temporal scales (Sagarin and Pauchard, 2009; Amici, 2011).

In view of these requirements, remote sensing represents a crucial source of information for measuring ecological complexity for several reasons, including: i) availability at multiple spatial scales (grain, pixel size) at the same time, ii) high temporal resolution, iii) coverage of large areas within relatively short timespans (Wegmann et al., 2014). As an example, remote sensing data have long been used for ecological applications such as biodiversity estimation, ecosystem management, restoration, hydrological modelling, land use mapping and climate change detection (e.g. Skidmore et al. (2015)).

Land and water resources managers around the world can now observe shifts in landscapes, nightscapes and waterscapes (Venot et al., 2007; Molle et al., 2012; Marcantonio et al., 2015) by combining remote sensing with spatio-temporal modeling (McCartney and Arranz, 2007; Ali et al., 2014). It is particularly important to monitor those resource constraints which can generate pressure on ecosystem services from various anthropogenic actors (Molle et al., 2012). Many software packages attempt to evaluate patterns of land use change and its impacts on land- and waterscapes (Baker et al., 1991; Rubin et al., 2003), and some of these packages consider long term dynamics (Coulthard, 2001).

A review of the field shows some independent specialized software and some integrated software, such as OSSIM, Orfeo ToolBox, Opticks, and

GRASS GIS. There is a growing demand from the scientific community as well as public and funding bodies for full reproducibility in research, and producing the exact set of code and data used in a research goes a long way towards permitting both peer-review and future research (Chemin et al., 2015). Reproducibility and robustness of software algorithms are two fundamental requirements to allow the continuity of scientific methods over time (Petras et al., 2015).

In this paper we will summarize the most straightforward measures of spatial complexity available in the Free and Open Source Software GRASS GIS, and relate them to the potential estimation of key ecological patterns and processes.

2 GRASS GIS based algorithms for complexity measurement from remote sensing

2.1 Why GRASS GIS?

GRASS GIS (Geographical Resources Analysis Support System, Neteler et al. (2012)) was first developed by the U.S. Army Construction Engineering Research Laboratories in the eighties. It allows managing and analyzing geographical data by 500 dedicated modules.

Worldwide contributions from the scientific community based on a free open source software (FOSS) license, available from 1999, and on an online source code repository (Concurrent Versioning System at that time) renders GRASS GIS one of the most cutting-edge projects of the Open Source Geospatial Foundation (OSGeo, founded in 2006).

In this research we will describe and illustrate the most powerful modules in GRASS GIS to measure spatial complexity from an ecological perspective. The methods are applicable to any raster imagery, but in ecology the datasets which are most commonly processed in these contexts are digital elevation models, categorical land-use maps or continuously-valued imagery derived from remote sensing, representing variables such as vegetation density.

We will make use of the free dataset called "North Carolina" available online at
http://grass.osgeo.org/download/sample-data/ together with additional Landsat ETM+ data, using GRASS GIS version 7.0.

2.2 Geometrical complexity: detecting edges

Geometrical complexity is a landscape property which is used as one of the main heuristics to distinguish individual patches by objective methods. Patches may be identified by detecting edges at different spatial scales under a hierarchical criterion (Burnett and Blaschke, 2003).

Current Object Based Image Analysis (OBIA) techniques generally build on edge detection (Thomas, 2010). In this section we will illustrate the most powerful techniques available in GRASS GIS to detect edges relying on: i) zero-crossing edge detection, ii) building vector contours from raster maps, iii) edge density and contrast weighted edge density calculation, iv) Canny filtering, v) Hough transforms.

2.2.1 Zero-crossing "edge detection" raster function for image processing: the i.zc function

The `i.zc` function allows users to locate boundaries using the zero-crossing algorithm based on the following arguments:

```
i.zc input=string output=string [width=integer]
[threshold=float] [orientations=integer]
```

where an `input` raster is converted to a zero-crossing raster map (`output`) with a specified Gaussian filter dimension (default is 9, but it can be changed by the argument `width`) and sensitivity (default is 10, but it can be changed by the argument `threshold`, together with the optional specification of the number of azimuth directions to be categorized (optional parameter `orientations`, default equals 1). Notice that, according to GRASS notation, arguments in square brackets are optional.

The procedure to find the edges in the image is based on the calculation of the Fourier transform of the image (see e.g. Rocchini et al. (2013b)) and the application of a Laplacian filter. The image is further processed, searching for local changes from positive to negative values. Where the change value crosses zero with respect to a defined threshold the pixel is marked as an edge.

As an example, using a Landsat7 ETM+ band as input, the output crossing edges are derived using the command shown below:

```
i.zc lsat7_2002_40 output=lsat7_2002_40_zerocrossing
```

leading to the output shown in Figure 1.

2.2.2 Producing a vector map of specified contours from a raster map by `r.contour`

In some cases, edge detection relates to linear objects in the imagery that are defined by a series of points having similar properties, e.g. the same elevation. As an example, generating contours from an input raster map is done using the `r.contour` function as shown below:

```
r.contour input=string output=string step=float
minlevel=float
maxlevel=float [levels=float[,float,...]]  [cut=integer]
```

where `input` and `output` are the original raster map and the output vector contours map respectively, `step` is the relative increment between adjacent contours values, `minlevel` and `maxlevel` are the minimum and maximum values in the image. These values can be derived using the function `r.info`.

As an example, let `elevation` be the input raster map; its contours might be derived simply as:

```
r.contour input=elevation output=elev_contours minlevel=50
maxlevel=160 step=10
```

producing the map shown in Figure 2.

2.2.3 Calculating edge density index on a raster map: `r.li.edgedensity`

Given a raster map, r.li.edgedensity is able to calculate a perimeter-to-area ratio, creating polygons based on a 4-neighbour rule. In the ecological context, such an approach is often applied to maps of land use in order to estimate the heterogeneity of the landscape and the fragmentation of its components.
The formula used is simply:

$$E = \frac{\sum(e_k)}{A} \times 10000 \tag{1}$$

where k=patch type and e_k=total edge length related to class k, A=total landscape area.

As in all the `r.li` functions in GRASS GIS, a configuration file (argument

`conf`) specifying the grain and the extent of analysis should be provided. This can be generated using the command `g.gui.rlisetup` which allows the user to choose the grain and extent of the calculation. In this paper we will rely on local moving windows *sensu* Hagen-Zanker (2016).
The final command is as follows:

```
r.li.edgedensity map=name conf=name output=name
```

2.2.4 Calculating contrast weighted edge density index on a raster map: `r.li.cwed`

By contrast with simple edge density, contrast weighting allows a weighting of the calculation based on:

$$CWED = \frac{\sum e_{ik} \times d_{ik}}{Area} \times 10000 \tag{2}$$

where k=attribute under consideration, e_{ik}=edge density between patch types i and k, d_{ik}=dissimilarity between patch types i and k, and Area=total landscape area.

In the ecological context, this varying dissimilarity is important because it allows certain types of boundary to be given more importance: for example, a boundary between hard surface and grassland represents more of a barrier to some dispersing species than a boundary between wet and dry grassland.

2.2.5 Canny edge detector

The `i.edge` function uses the edge detector defined by Canny (1986) to detect edges in a raster map. The Canny edge detector is considered optimal by Sonka et al. (1999) based on the following criteria: i) important edges cannot be omitted and only actual edges can be detected as edges, ii) the difference in position of the actual and the detected edge is minimal, iii) there is only one detected edge for an actual edge in the original image. The Canny edge detector first reduces the noise in the raster map using a Gaussian filter. Then it computes gradient defined by an angle and magnitude. The next step is non-maximum suppression, which preserves only those pixels with magnitude higher than magnitude of other pixels in the direction of the gradient. The final step extracts significant edges by thresholding with hysteresis. The Canny edge detector can be applied using the following command:

```
i.edge input=name output=name [angles_map=name]
[low_threshold=float] [high_threshold=float] [sigma=float]
```

where `input` is an image, `output` is a raster map containing the detected edges, `angles_map` is a raster map containing the angle of the image, `sigma` is the size of the moving window (kernel) used and `low_threshold` and `high_threshold` are used during the thresholding with hysteresis as follows: values over the `high_threshold` are kept; values under `low_threshold` are removed; values in between these constants are kept only when the pixel touches another pixel with value above `high_threshold`.

The result of `i.edge` function is a binary raster image where edges are represented as rasterised lines exactly one pixel wide. The detected edges can be used for further analysis using for example, the `r.neighbors` function which can extract areas with high or low edge density. In Figure 3, areas with many edges are associated with developed areas, while areas with low density indicate natural areas. The result can be used also as an input for a Hough transform.

2.2.6 Hough transform

The Hough transform is a feature extraction technique which identifies straight line segments from a raster image and outputs them as vector features. Such a technique is applicable to edges detected and rasterised using the methods described above (Hough, 1962; Duda and Hart, 1972). Points in the real space which are assumed to represent points on an edge are transformed into a Hough plane applying the following equation to describe a line:

$$x \cos \theta + y \sin \theta = r \tag{3}$$

where r is the length of a normal from the origin in the Hough plane to the line and θ is the angle of the normal.

Points in the original image which belong to one line result in sinusoidal curves intersecting in one point in the transformed image as in Figure 4. The coordinates of this point describe the parameters r and θ of the line, and its value represents the number of points on the line.

The `r.houghtransform` function in GRASS GIS uses the 'identify and remove' method proposed by Fiala (2003) which identifies the most prominent lines in a raster image and outputs the coordinates of the associated line segments. Galambos et al. (2000) showed that the detection is significantly faster when the gradient direction of the edge is provided as well. GRASS GIS uses this extension when the direction is available.

Using the Hough transform, GRASS GIS detects the linear features using the following:

```
r.houghtransform input=name output=name [angles=name]
[hough_image=name] [max_gap_count=integer]
[min_segment_length=integer]
```

where `input` is a raster map containing edges, `output` is a vector map containing detected straight line segments, `angles` is an optional input for speedup, `hough_image` is an optional output for visual inspection of the Hough transform, `max_gap_count` is a maximal allowed number of gaps in one line segment and `min_segment_length` is a minimal allowed length of one line segment. There are several other parameters which ensure fine control over the number and properties of the detected lines.

The typical input to a Hough transform is a raster image containing thin edges detected e.g. by the `i.edge` function. The straight (and, depending on configuration, more or less long) lines which result from the `r.houghtransform` function can be used as indicators of man-made features such as the straight parts of a highway visible in Figure 5. The `r.houghtransform` can be also applied to terrain or surface contours to retrieve straight lines in terrain, possibly associated with roads, buildings and other man-made structures. Furthermore, Hough transform can be used to automatically detect geological lineaments (Vasuki et al., 2014; Wang and Howarth, 1990).

2.3 Local diversity in a neighbourhood

Calculating local diversity is important to detect spots of diversity at a local scale. As an example, in biodiversity research, this is known as α-diversity and it is a widely-used metric in ecology (Rocchini et al., 2010).

2.3.1 Local statistics by `r.neighbors`

The `r.neighbors` command provides the means to compute a variety of local statistic, including: average, median, mode, minimum, maximum, range, standard deviation, sum, count, variance, diversity (i.e. the number of different values in the neighbourhood with respect to the central pixel), interspersion (weighted diversity), first quartile, third quartile, user-specified quantiles.

In the case where one is interested in a measure of complexity over space, standard deviation in a neighbourhood might be simply calculated as follows:

```
r.neighbors input=name output=name method=sttdev
[size=value]
```

The size may be changed to enlarge the window of analysis, starting with a default of 3×3 cells.

2.3.2 Information-theory based statistics: `r.li.shannon`, `r.li.pielou`, `r.li.simpson`, `r.li.renyi`

GRASS GIS is capable of handling common Information-theory based statistics such as Boltzman or Shannon-Weaver entropy H (Shannon, 1948), Pielou evenness (Pielou, 1966) and Simpson's reversed dominance (1-D, Simpson (1949)).

Different diversity measures are generally used to summarise large multivariate data sets, providing for one potentially meaningful single value. Such an approach inevitably results in information loss, since no single summary statistic can characterize in an unequivocal manner all aspects of diversity (Ricotta, 2005; Marcantonio et al., 2014). Rocchini and Neteler (2012) addressed such problems when measuring diversity from a satellite image relying on the richness and relative abundance of Digital Numbers (DNs), by only using entropy-based metrics. In particular, they observed: i) the intrinsic impossibility of discriminating among different ecological situations with one single diversity index, and ii) the impossibility of understanding whether diversity of different sites is more related to differences in richness or in relative abundance of DN values. As an example, they provided a theoretical case in which the same value of the Shannon index would actually be related to very different situations in terms of DNs richness and abundances (see Figure 2 in Rocchini and Neteler (2012)). In general, to solve this issue, combining these entropy-based indices with evenness-based metrics might lead to an increase in their information content. In this regard, the most commonly-used metric is the Pielou evenness index $J = \frac{-\sum p \times ln(p)}{ln(N)}$ **(Pielou, 1969)**, which can be rewritten as: $J = \frac{H}{Hmax}$ since it contains the maximum possible diversity (ln(N)), for N DNs.

All the previously described metrics based on Information theory only supply point descriptions of diversity. By contrast, Rényi (1970) firstly introduced a generalized entropy metric, $H_\alpha = \frac{1}{1-\alpha} \times ln \sum p^\alpha$ which shows a high flexibility and power because a number of popular diversity indices are special cases of H_α. In mathematical terms, if we consider e.g the variation of α from 0 to 2:

$$H\alpha = \begin{cases} \alpha = 0, H_0 = ln(N) \\ \alpha = 1, H_1 = -\sum(p \times ln(p)) \\ \alpha = 2, H_2 = ln(1/D) \end{cases} \tag{4}$$

where N = number of Digital Numbers (DNs), p = relative abundance of each DN value, D = Simpson index.

Concerning the results attained when alpha=1, the Shannon index is derived according to the L'Hôpital's rule of calculus (see Ricotta (2005). Rényi generalized entropy represents a continuum of diversity measures Ricotta and Avena (2003)), meaning that it is possible to maintain sensitivity to both rare and abundant DNs, and it is more responsive to the commonest DNs while α increases. Varying α can be viewed as a scaling operation, not in a real space but in the data space.

As far as we know, GRASS GIS is the only software capable of calculating generalized measures of diversity such as the Rényi formula in a 2-dimensional space, based on the following function:

```
r.li.renyi conf=conf3 in=landsat.pc1 out=landsatrenyi
alpha=2
```

Changing the parameter α will change the behaviour of the formula, generating different maps of diversity as represented in Figure 6, representing a continuum of diversity values over space instead of single measures. Increasing alpha values in the Rényi diversity index will weight differences in relative abundance more heavily than differences in simple richness.

2.4 Texture-based metrics (*sensu* Haralick et al. (1973))

2.4.1 Generating images with textural features from a raster map: `r.texture`

GRASS GIS permits computation of all the local textural features that may be calculated in a neighborhood of pixels, described in the benchmark paper by Haralick et al. (1973): i) the angular second moment, as a measure of local homogeneity; ii) the contrast, a gray-level variation with respect to neighbor pixels; iii) the correlation, a linear dependency value; iv) the variance in the neighboring moving window (see also `r.neighbors`); v) the entropy, an index of randomness; vi) the sum average; vii) the sum entropy; viii) the sum variance; ix) the difference in variance; x) the difference in entropy; xi) the inverse distance moment, i.e. the inverse of the previously described contrast measure; xii) the maximal correlation coefficient. We refer to Haralick et al. (1973) for a detailed description of all the measures.

The approach to be used can be declared as the `method` parameter of the function `r.texture`, as follows:

```
r.texture input=landsat.pc1 method=asm,contrast,corr,var,idm,
sa,se,sv,entr,dv,de,moc1,moc2 output=texture
```

Figure 7 presents all the aforementioned maps generated from a Landsat ETM+ image.

Further, the following R code can show the amount of correlation among different measures once data are imported in R by the `rgrass7` package, as shown below:

```
# require the rgrass7 library to import GRASS data in R
require(rgrass7)
# import data textureset <- readRAST(c("texture_ASM",
"texture_Contr","texture_Corr",
"texture_Var",
"texture_Entr","texture_SA","texture_SE","texture_SV",
"texture_DV", "texture_DE", "texture_IDM",
"texture_MOC−1"), cat=c(F,F,F,F,F,F,F,F,F,F,F,F))
# require the hexbin package to do an hexagon binning between
variables
hbin <- hexbin(textureset$texture_IDM,
textureset$texture_Contr, xbins=50)
plot(hbin)
```

Figure 8 shows the correlation trends found applying this code, while the hexagon binning plots are shown in the Supplementary Material of this manuscript. The majority of the variables were strongly correlated (Figure 9, generated by the `corrplot` package in R), showing the high multicollinearity of the texture measures system. Once such relations are used to plot maps derived from each other, the similarity is apparent. Figure 10 shows the map of estimated Sum Entropy from Entropy (by applying a linear model, R^2=0.9023, $p<0.001$) which is similar to the original one, while residuals distribution follows, as expected, the magnitude of the values of the predicted variable. Hence, when modelling ecosystem complexity, texture measures should be used with care since, by their very nature, they are expected to be correlated with each other.

2.5 Detecting heterogeneity in synthetic spaces

2.5.1 Fast Fourier Transforms (FFT) for image processing: `i.fft`

The use of transforms in frequency spaces to measure variation in a signal has long been acknowledged. While methods exist based on orthonormal series (e.g. rectangular decomposition of waves, Walsh (1923), the most commonly-used methods rely on continuous waves, mainly based on the Fourier transforms (Fourier, 1822).

When seeking a method to detect landscape change based on continuous instead of classified information, one should rely on a (continuous) function which does not require a) a-priori field information nor ii) a specific model based on the data being used. In view of this, Fourier transforms (Fourier, 1822) may represent the best algorithmic solution.
Let f(x) be a continuous function described into a spatial domain. Based on the Fourier theorem (Fourier, 1822) every f(x) can be transformed into a continuum of sinusoidal functions of varying frequency, as follows:

$$F(\omega) = \int_{-\infty}^{\infty} f(x) \mathrm{e}^{-2\pi i \omega} \, \mathrm{d}x \tag{5}$$

where ω = frequency, also known as radian frequency since it is expressed in radians per spatial units. In mathematical notation for discrete Fourier transforms f(x)F(ω). Extending Eq. (4) to two dimensions implies considering a two-dimensional function f(x,y), e.g. a raster matrix. Its Fourier transform turns out to be:

$$F(\omega, \nu) = \int \int_{-\infty}^{\infty} f(x, y) \mathrm{e}^{-2\pi i(\omega x + \nu y)} \, \mathrm{d}x, \mathrm{d}y \tag{6}$$

where ω,ν= frequency coordinates.
Considering as an example a single raster image (e.g. the first Principal Component of a Landsat scene) the command to be used to calculate its Fourier transform is straightforward:

```
i.fft input_image=lsat_pca1 real=lsat_pca1_real
imaginary=lsat_pca1_imag
```

where `real`=real part of Eq.(4) and Eq.(5) and `imaginary`=imaginary part of Eq.(4) and Eq.(5), both stored as raster maps. An example of the output is provided in Figure 11.
In the Fourier space, high frequency values (high heterogeneity) are at the border of the image while low frequency values (high homogeneity) are at the center. Hence the higher the value of pixels at the border, the higher the heterogeneity / complexity of the whole image.

2.6 Testing complexity against random surfaces

Observed ecological patterns can be tested again random patterns by calculating the deviation from random expectations in two dimensions (Hanspach et al., 2011). To accomplish this goal, different kinds of lattice surfaces can be generated, including: completely random surfaces, gaussian distributed and fractal surfaces with a predefined fractal dimension.

2.6.1 Generating random surfaces by `r.random.surface`

Random surfaces can be generated by the following basic function and arguments command:

```
r.random.surface output=string [distance=value]
[exponent=value]
```

where `distance` represents the maximum distance of spatial correlation among pixels and `exponent` represents the exponential decay of values over space. As an example, Figure 12 represents a random surface generated by the aforementioned command. As an example, a Landsat image might be tested against this to find areas where similar values are especially clumped and significantly deviate from random expectations over space.

2.6.2 Generating gaussian random number maps by `r.surf.gauss`

A more sophisticated but still straightfoward neutral model is represented by a surface whose values have a normal distribution in two dimensions.
This can be created by the following command:

```
r.surf.gauss output=name [mean=value] [sigma=value]
```

where the mean and the standard deviation (σ) can be defined a-priori (Figure 12).

2.6.3 Neutral landscapes by fractal surfaces of a given fractal dimension: `r.surf.fractal`

Following Mandelbrot (2006), surfaces with a given fractal dimension from 2 to 3 might represent severe differences in their roughness / complexity (Imre et al., 2011). Such surfaces can be generated in GRASS GIS by the function `r.surf.fractal` by explicitly stating the fractal dimension according to the parameter `dimension`, as:

```
r.surf.fractal output=name [dimension=value]
[number=value]
```

A very useful parameter is represented by `number` which indicates the number of intermediate surfaces one might want to generate to finally gather a complete set of images of variable fractal dimension (Figure 12).

3 Summary of the presented algorithms

As described in this paper, there are many ways of defining complexity (Anand and Tucker, 2003), and then measuring it. Every single measure of complexity has a potential spatio-ecological application, in particular when it is applied to remotely sensed imagery: from feature extraction by edge detection (Zhang et al., 2005), to biodiversity estimation by information theory (e.g. Rocchini et al. (2010)), to forest structure variable retrieval by textural analysis (Kayitakire et al., 2006), and multifractal surfaces generated as neutral models for e.g. feature change assessment (Cheng, 1999).

We structured our paper to consider all the different aspects of complexity in a variety of potential spatial fields of research: from geometrical complexity to information theory-based measures, to texture, reprojected spaces and random surfaces. In this paper we have accounted only for spatial complexity, while ecological dynamics (temporal complexity) might be further studied using throughput analytic approaches based on e.g. i) stationary Markov models (Tucker and Anand, 2005), ii) Monte Carlo analysis of multitemporal series (Van Niel et al., 2005), or iii) Kohonen neural networks (Foody and Cutler, 2006). The present paper mainly aims to describe features that are already implemented in the GRASS GIS platform rather than describing the procedure to implement new features. It can be stated that GRASS GIS offers a concrete possibility of implementing new features rather easily using its collection of excellent internal and external software libraries.

GRASS GIS offers the tools to compute a number of pre-existing measures of complexity, as well as the possibility to generate and evaluate new ones, because of the free and open access to the source code. The modular software design of GRASS facilitates the introduction and sharing of new functionalities without affecting the overall performance of the system. Moreover, its scripting capabilities enable automated processing of a large volume of data and wide-ranging use of the achieved results. In particular, recent developments also allow GRASS users and developers to make use of the Python programming language (Van Rossum (1995)) to introduce new features.

4 Acknowledgements

We are grateful to the Handling Editor E. Pebesma and two anonymous reviewers who provided useful insights of a previous version of this paper.

References

Ali, S., Mehmood, H., Chemin, Y., 2014. Bmp implementations in himalayan context: Can a locally-calibrated swat assessment direct efforts? International Journal of Geoinformatics 10, 53–62.

Amici, V., 2011. Dealing with vagueness in complex forest landscapes: A soft classification approach through a niche-based distribution model. Ecological Informatics 6, 371–383.

Anand, M., Tucker, B., 2003. Defining biocomplexity - an ecological perspective. Comments on Theoretical Biology 8, 497–510.

Baker, W. L., Egbert, S. L., Frazier, G. F., 1991. A spatial model for studying the effects of climatic change on the structure of landscapes subject to large disturbances. Ecological Modelling 56, 109–125.

Burnett, C., Blaschke, T., 2003. A multi-scale segmentation/object relationship modelling methodology for landscape analysis. Ecological Modelling 168, 233–249.

Canny, J., Jun. 1986. A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. 8, 679–698.

Chemin, Y., Petras, V., Petrasova, A., Landa, M., Gebbert, S., Zambelli, P., Neteler, M., Löwe, P., Di Leo, M., 2015. GRASS GIS: a peer-reviewed scientific platform and future research repository. In: Geophysical Research Abstracts. Vol. 17. p. 8314.

Cheng, Q., 1999. The gliding box method for multifractal modeling. Computers & Geosciences 25, 1073–1079.

Coulthard, T. J., 2001. Landscape evolution models: a software review. Hydrological processes 15, 165–173.

Duda, R. O., Hart, P. E., 1972. Use of the hough transformation to detect lines and curves in pictures. Communications of the ACM 15, 11–15.

Fiala, M., 2003. Identify and remove hough transform method. In: Proc. Vision Interface. pp. 184–187.

Foody, G., Cutler, M., 2006. Mapping the species richness and composition of tropical forests from remotely sensed data with neural networks. Ecological Modelling 195, 37–42.

Fourier, J. B. J., 1822. Théorie Analytique De La Chaleur. Didot.

Galambos, C., Kittler, J., Matas, J., 2000. Using gradient information to enhance the progressive probabilistic hough transform. In: Pattern Recognition, 2000. Proceedings. 15th International Conference on. Vol. 3. IEEE, pp. 560–563.

Hagen-Zanker, A., 2016. A computational framework for generalized moving windows and its application to landscape pattern analysis. International Journal of Applied Earth Observation and Geoinformation 44, 205–216.

Hanspach, J., Kühn, I., Schweiger, O., Pompe, S., Klotz, S., 2011. Geographical patterns in prediction errors of species distribution models. Global Ecology and Biogeography 20, 779–788.

Haralick, R., Shanmugam, K., Dinstein, I., 1973. Textural features for image classification. IEEE Transactions on Systems, Man and Cybernetics SMC-3, 610–621.

Hough, P., 1962. Method and means for recognizing complex patterns. U.S. Patent 3, 0069.654.

Imre, A. R., Cseh, D., Neteler, M., Rocchini, D., 2011. Korcak dimension as a novel indicator of landscape fragmentation and re-forestation. Ecological Indicators 11, 1134–1138.

Kayitakire, F., Hamel, C., Defourny, P., 2006. Retrieving forest structure variables based on image texture analysis and IKONOS-2 imagery. Remote Sensing of Environment 102, 390–401.

Mandelbrot, B., 2006. Fractal geometry: what is it, and what does it do? Proceedings of the Royal Society A: Mathematical and Physical Sciences 423, 3–16.

Marcantonio, M., Pareeth, S., Rocchini, D., Metz, M., Garzon-Lopez, C. X., Neteler, M., Jun. 2015. The integration of Artificial Night-Time Lights in landscape ecology: A remote sensing approach. Ecological Complexity 22, 109–120.

Marcantonio, M., Rocchini, D., Ottaviani, G., Jul. 2014. Impact of alien species on dune systems: a multifaceted approach. Biodiversity and Conservation, 1–24.

McCartney, M. P., Arranz, R., 2007. Evaluation of historic, current and future water demand in the Olifants River Catchment, South Africa. Vol. 118. IWMI.

Molle, F., Foran, T., Kakonen, M., 2012. Contested waterscapes in the Mekong region: Hydropower, livelihoods and governance. Earthscan.

Neteler, M., Bowman, M. H., Landa, M., Metz, M., 2012. Grass gis: A multi-purpose open source gis. Environmental Modelling & Software 31, 124–130.

Petras, V., Petrasova, A., Harmon, B., Meentemeyer, R., Mitasova, H., Jun 2015. Integrating free and open source solutions into geospatial science education. ISPRS International Journal of Geo-Information 4, 942–956.

Pielou, E., 1966. The measurement of diversity in different types of biological collections. Journal of Theoretical Biology 13, 131–144.

Ricotta, C., 2005. Additive partitioning of rao's quadratic diversity: a hierarchical approach. Ecological Modelling 183, 371, 365.

Ricotta, C., Avena, G., 2003. On the relationship between pielou's evenness and landscape dominance within the context of hill's diversity profiles. Ecological Indicators 2, 361–365.

Rocchini, D., Balkenhol, N., Carter, G. A., Foody, G. M., Gillespie, T. W., He, K. S., Kark, S., Levin, N., Lucas, K., Luoto, M., Nagendra, H., Oldeland, J., Ricotta, C., Southworth, J., Neteler, M., 2010. Remotely sensed spectral heterogeneity as a proxy of species diversity: Recent advances and open challenges. Ecological Informatics 5, 318–329.

Rocchini, D., Foody, G. M., Nagendra, H., Ricotta, C., Madhur, He, K. S., Amici, V., Kleinschmit, B., Forster, M., Schmidtlein, S., Feilhauer, H., Ghisla, A., Metz, M., Neteler, M., 2013a. Uncertainty in ecosystem mapping by remote sensing. Computers & Geosciences 50, 128–135.

Rocchini, D., Metz, M., Ricotta, C., Landa, M., Frigeri, A., Neteler, M., 2013b. Fourier transforms for detecting multitemporal landscape fragmentation by remote sensing. International Journal of Remote Sensing 34, 8907–8916.

Rocchini, D., Neteler, M., 2012. Spectral rank-abundance for measuring landscape diversity. International Journal of Remote Sensing 33, 4458–4470.

Rubin, E., Shaffer, C., Ramakrishnan, N., Watson, L., Dymond, R., Kibler, D., Dietz, R., Chanat, J., Lohani, V., Bosch, D., Speir, C., 2003. From landscapes to waterscapes: A pse for landuse change analysis. Engineering with Computers 19, 9–25.

Sagarin, R., Pauchard, A., 2009. Observational approaches in ecology open new ground in a changing world. Frontiers in Ecology and the Environment 8, 379–386.

Shannon, C., 1948. A mathematical theory of communication. Bell System Technical Journal 27, 379–423.

Simpson, E., 1949. Measurement of diversity. Nature 163, 688.

Skidmore, A., Pettorelli, N., Coops, N., Geller, G.N., H. M., Lucas, R., Mücher, C., O'Connor, B., M., P., Pereira, H., Schaepman, M., Turner, W., Wang, T., Wegmann, M., 2015. Environmental science: Agree on biodiversity metrics to track from space. Nature 523, 403–405.

Sonka, M., Hlavac, V., Boyle, R., 1999. Image processing, analysis, and machine vision. PWS Pub.

Thomas, B., 2010. Object based image analysis for remote sensing. ISPRS Journal of Photogrammetry and Remote Sensing 65, 2–16.

Tucker, B. C., Anand, M., 2005. On the use of stationary versus hidden markov models to detect simple versus complex ecological dynamics. Ecological Modelling 185, 177–193.

Van Niel, T., McVicar, T., Datt, B., 2005. On the relationship between training sample size and data dimensionality: Monte carlo analysis of broadband multi-temporal classification. Remote Sensing of Environment 98, 468–480.

Van Rossum, G., 1995. Python Library Reference. CWI Report CS-R9524.

Vasuki, Y., Holden, E.-J., Kovesi, P., Micklethwaite, S., 2014. Semi-automatic mapping of geological structures using uav-based photogrammetric data: An image analysis approach. Computers & Geosciences 69, 22–32.

Venot, J.-P., Turral, H., Samad, M., Molle, F., 2007. Shifting waterscapes: explaining basin closure in the Lower Krishna Basin, South India. Vol. 121. IWMI.

Walsh, J., 1923. A closed set of orthonormal functions. American Journal of Math 45, 5–24.

Wang, J., Howarth, P. J., July 1990. Use of the hough transform in automated lineament. IEEE Transactions on Geoscience and Remote Sensing 28 (4), 561–567.

Wegmann, M., Santini, L., Leutner, B., Safi, K., Rocchini, D., Bevanda, M., Latifi, H., Dech, S., Rondinini, C., 2014. Role of african protected areas in maintaining connectivity for large mammals. Philosophical Transactions of the Royal Society of London B: Biological Sciences 369.

Zhang, Q., Pavlic, G., Chen, W., Fraser, R., Leblanc, S., Cihlar, J., 2005. A semi-automatic segmentation procedure for feature extraction in remotely sensed imagery. Computers & Geosciences 31, 289–296.

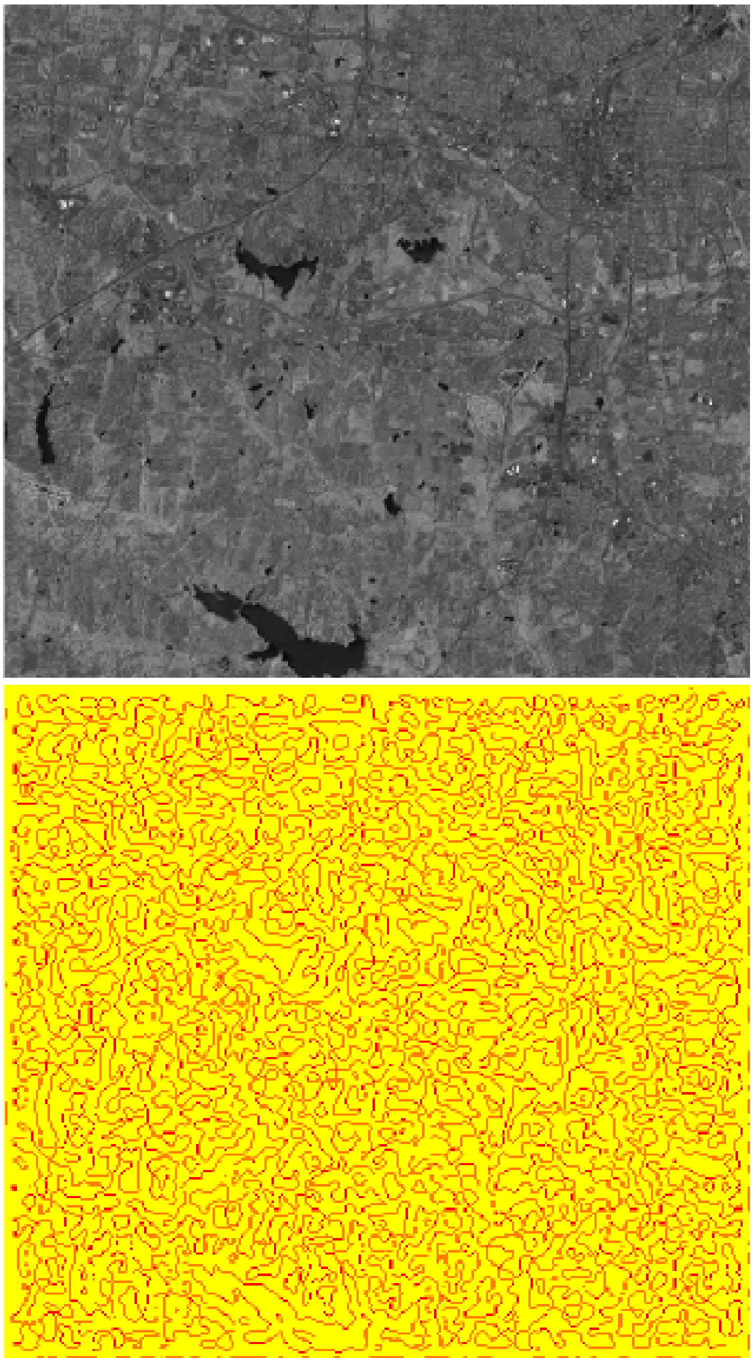


Figure 1: Zero-crossing "edge detection" raster function for image processing. A Landsat ETM+ band (near infrared) is processed and edges are revealed thanks to the `i.zc` function in GRASS GIS. Refer to the main text for additional information.

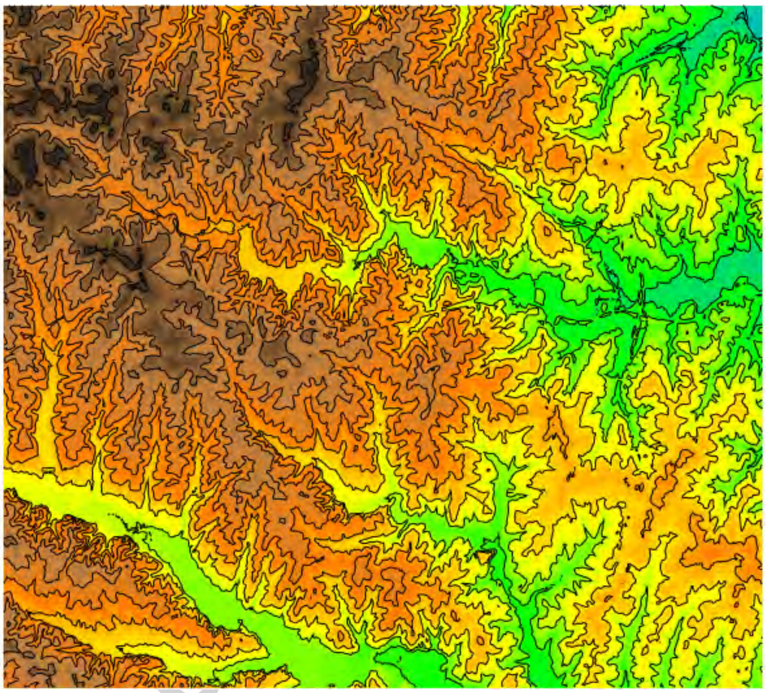


Figure 2: An elevation map from the GRASS North Carolina free dataset showing an elevation map and its contour with a step of 10 meters.

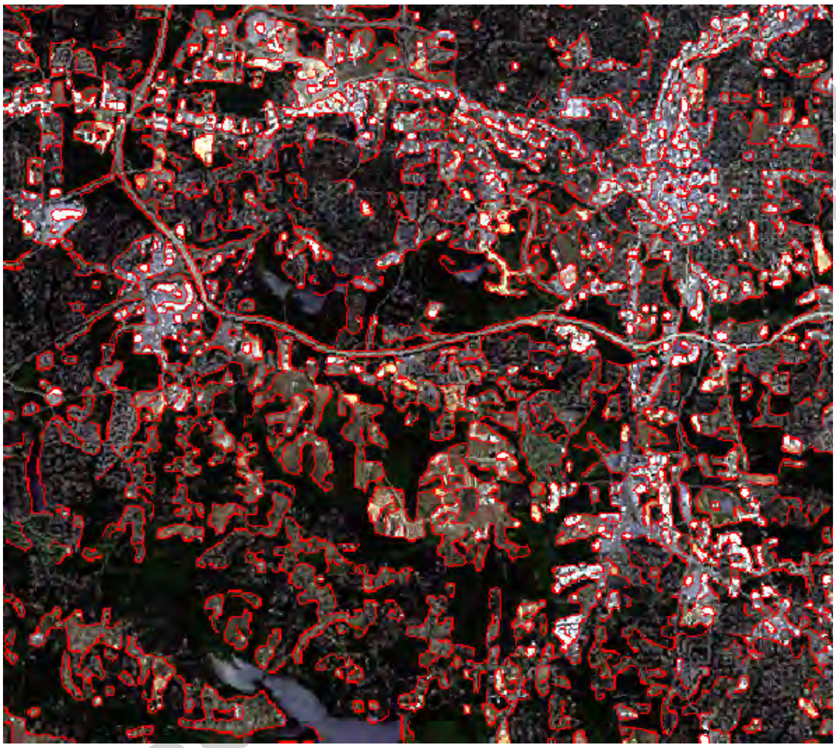


Figure 3: Edges (in red) detected by Canny edge detector on first component from PCA computed on 9 channels from Landsat 7, 2002, RGB channels in the background.

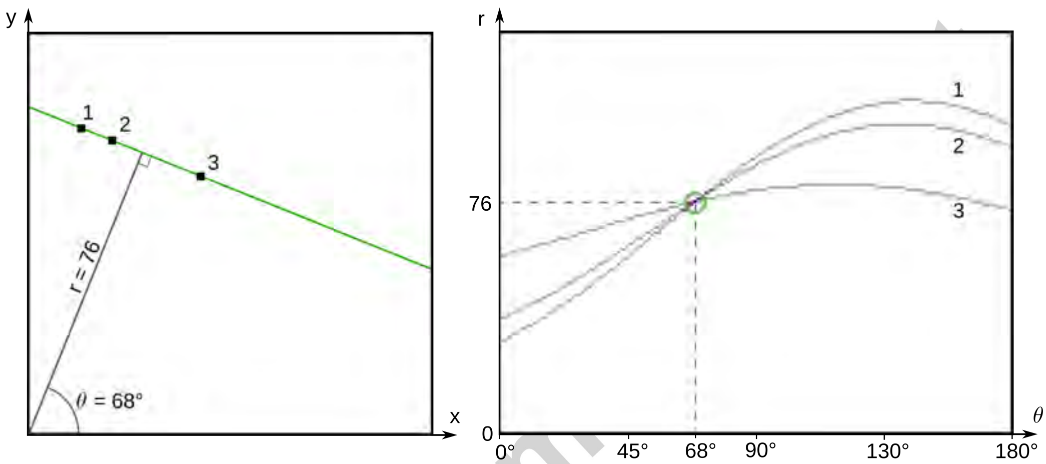


Figure 4: The linear feature to be automatically detected lies on the green line in the left image (coordinate space defined by {x,y}) and is represented only by several points (black pixels). Each point in the left image is transformed into a curve in the right image (coordinate space defined by {θ,r}) by considering lines in all directions θ passing through the point. The coordinates of the intersection of the curves in the right image are the parameters of the line in the left image.

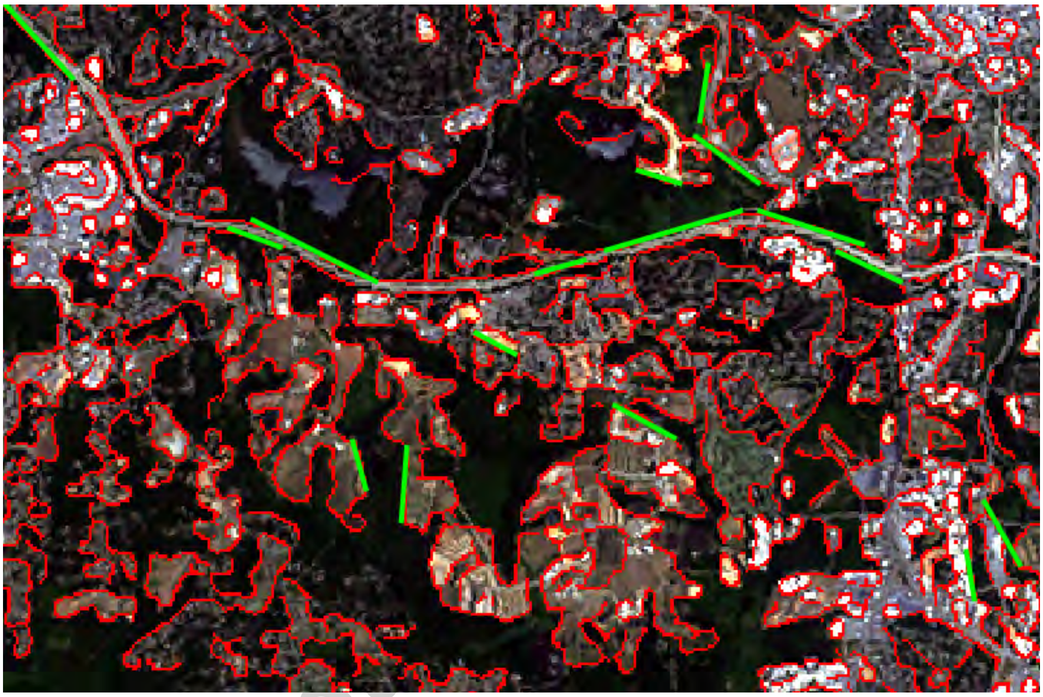


Figure 5: Detail of the central area from Figure 3 with lines obtained through Hough transformation (green) computed using the edges from Canny detector (red). Only the long lines, especially straight portions of the road, are detected.

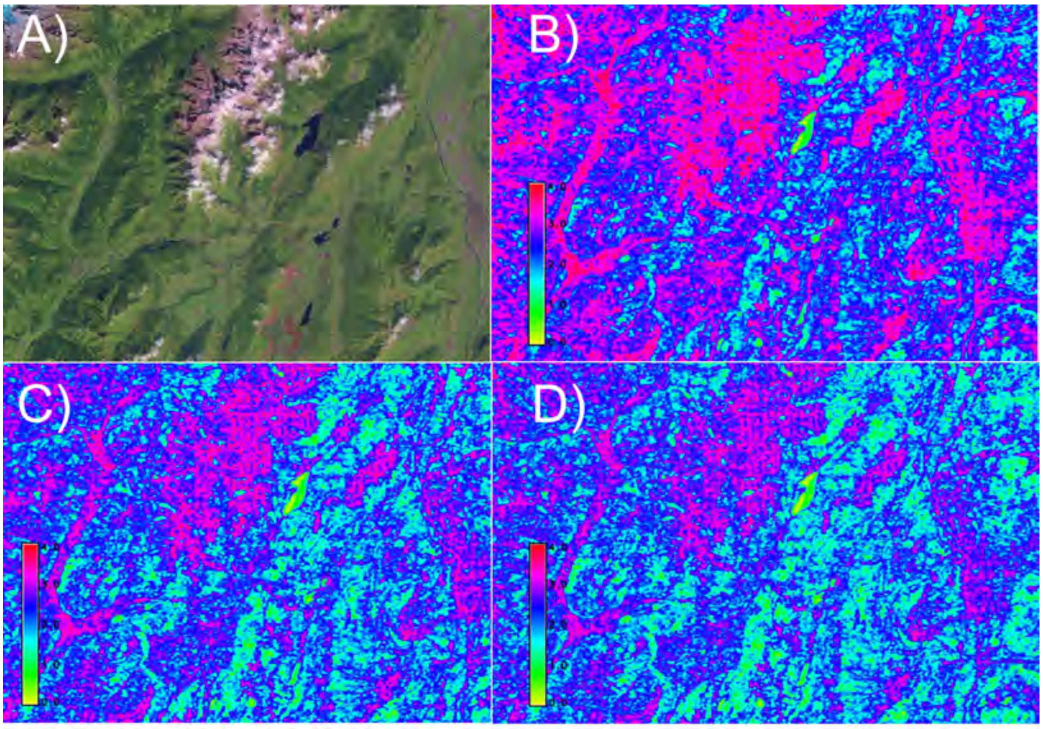


Figure 6: Rényi entropy can be calculated into GRASS 7.0 by the `r.li.renyi` command. In this example, starting from a Landsat ETM+ image, or a derivative like the first Principal Component, one might calculate different maps of Rényi entropy with different α values according to the formula $H_\alpha = \frac{1}{1-\alpha} \times ln \sum p^\alpha$. In this case α=2 (B), α=5 (C), α=7 (D). Refer to the main text for additional information.

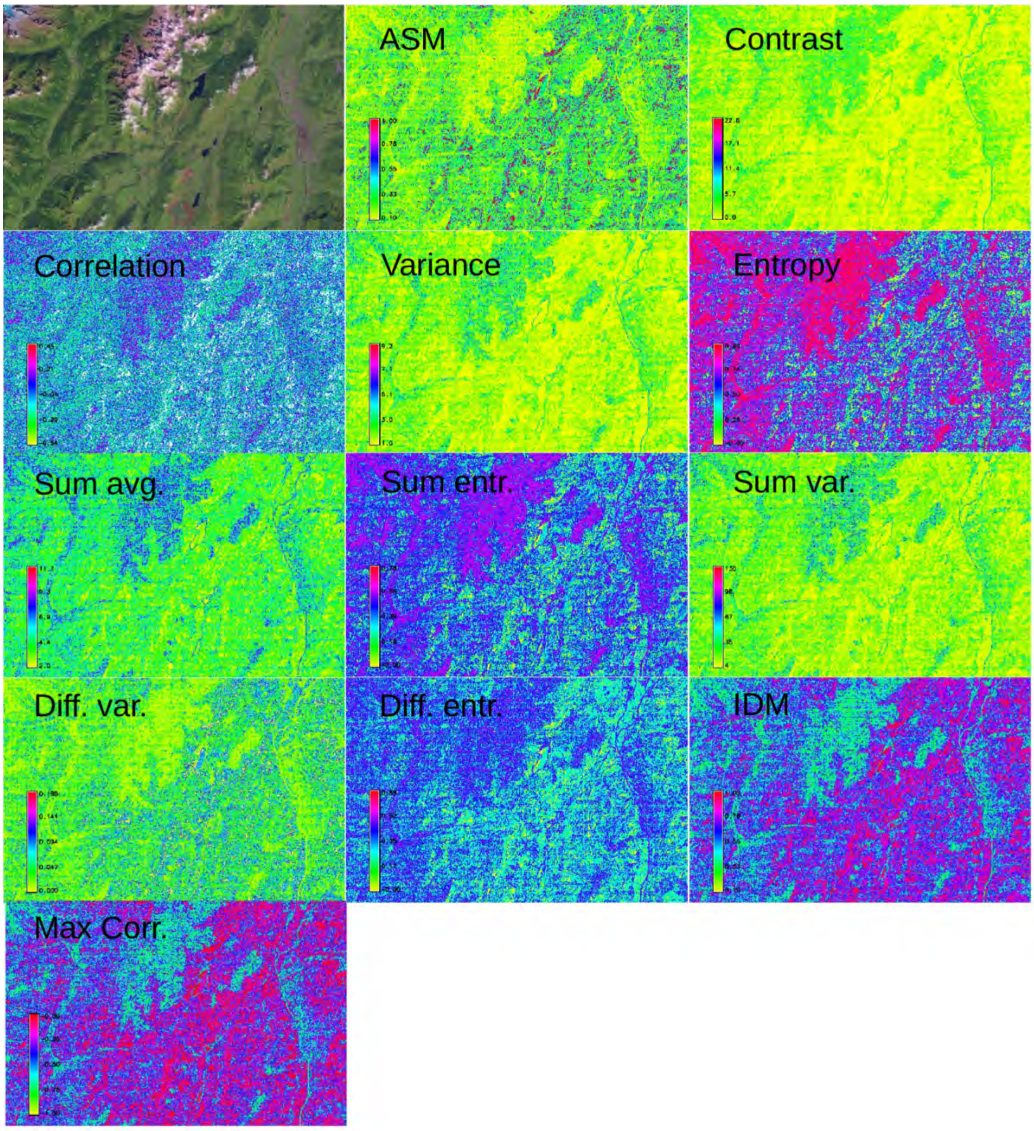


Figure 7: Different measures of texture as described in Haralick et al. (1973) starting from a Landsat ETM+ image of the Trentino region (Northern Italy). Acronyms: ASM = Angular Second Moment; IDM = Inverse Distance Moment

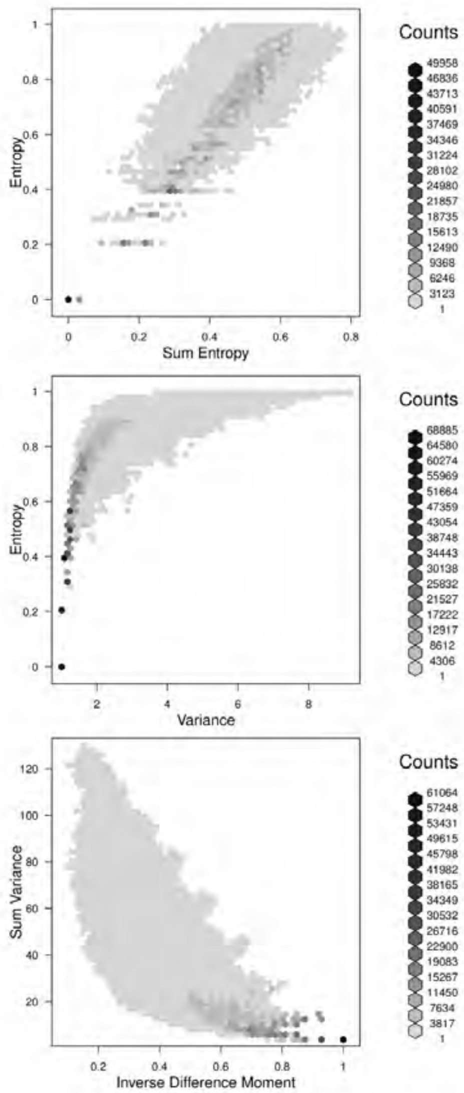


Figure 8: Hexagon binning showing the muticollinearity of a set of texture measures. Some of the main trends found, from top to bottom: linear relationship, power positive relationship, exponential decay. All the hexagon binning plots among the measured texture variables are available as Supplementary Material of this manuscript. If such variables are further used as predictors in e.g. a multiple regression model as complexity variables, they might be used with care since they basically carry the same (inverse, in this case) information.

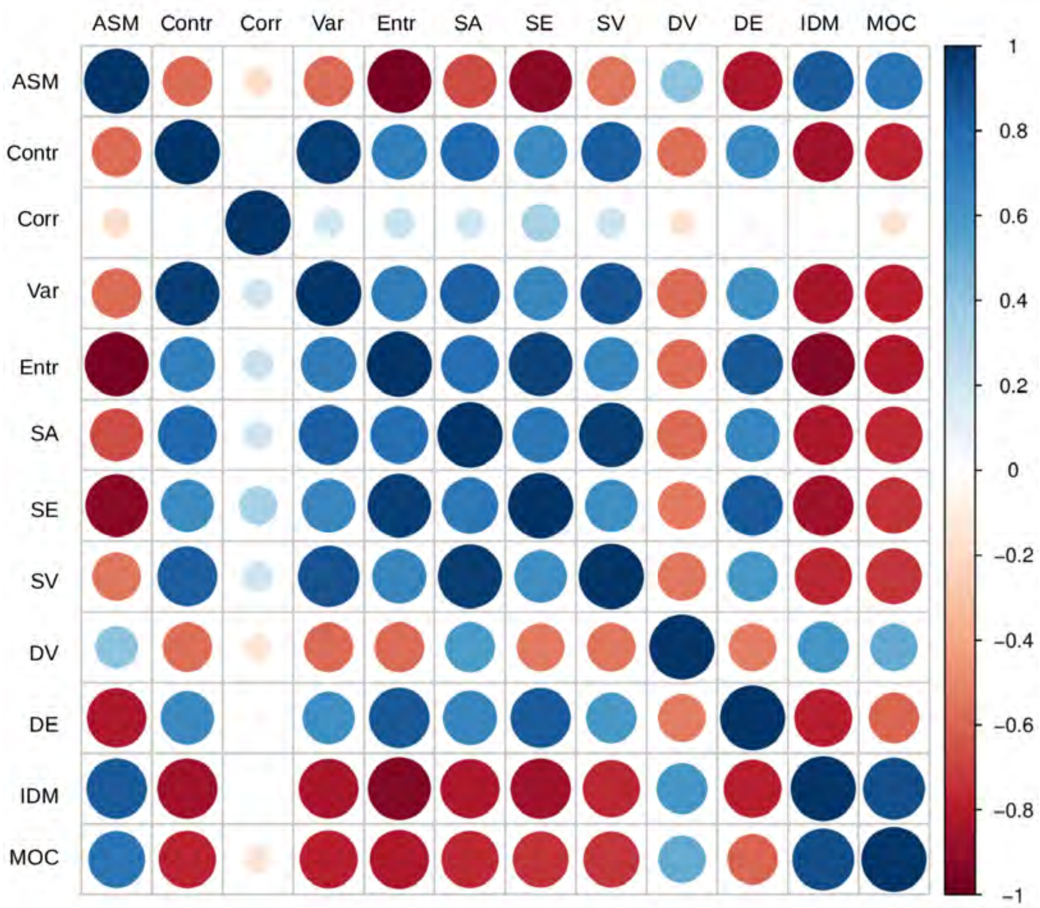


Figure 9: Correlations among the texture variables measured in GRASS GIS on a Landsat ETM+ of the Trentino region (Northern Italy), see Haralick et al. (1973), generated by the `corrplot` package in R. Only few variables showed a correlation near zero while most of them showed a high pairwise positive or negative correlation, demonstrating the basic multicollinearity of the texture measures system. ASM = Angular Second Moment, Contr = Contrast, Corr = Correlation, Var = Variance, Entr = Entropy, SA = Sum Average, SE = Sum Entropy, SV = Sum Variance, DV = Difference Variance, DE = Difference Entropy , IDM = Inverse Difference Moment, MOC = Information Measures of Correlation

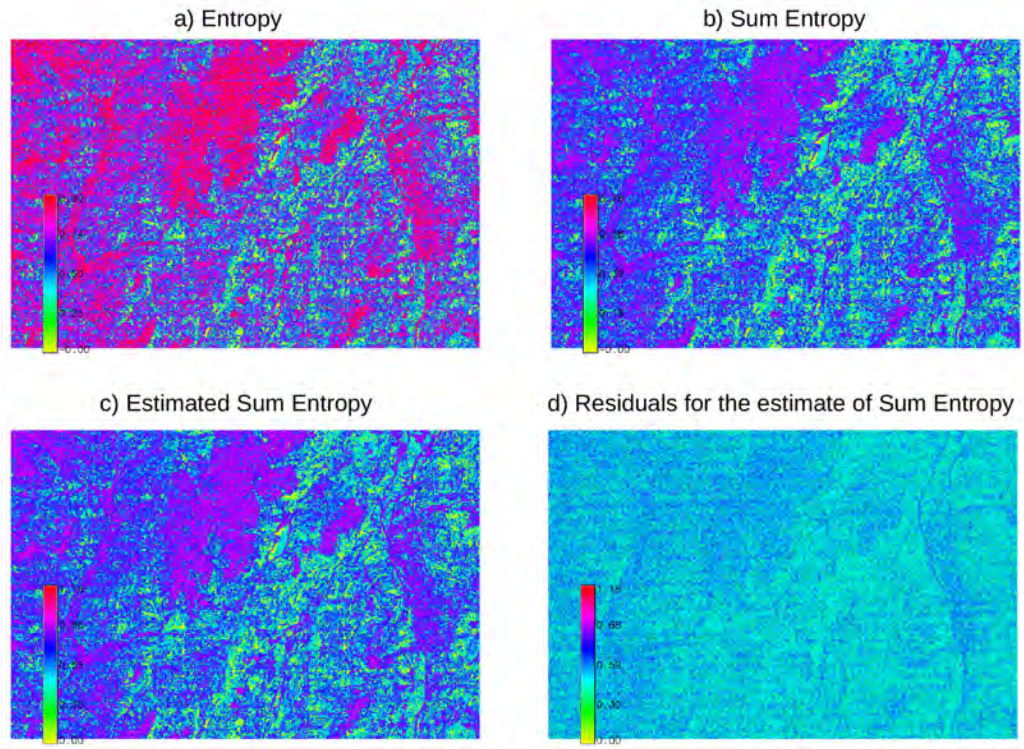


Figure 10: sExample of the estimated values of a texture variable starting from another one. In this case, Sum Entropy is estimated from the Entropy variable, showing a similar pattern of the original Sum entropy image.

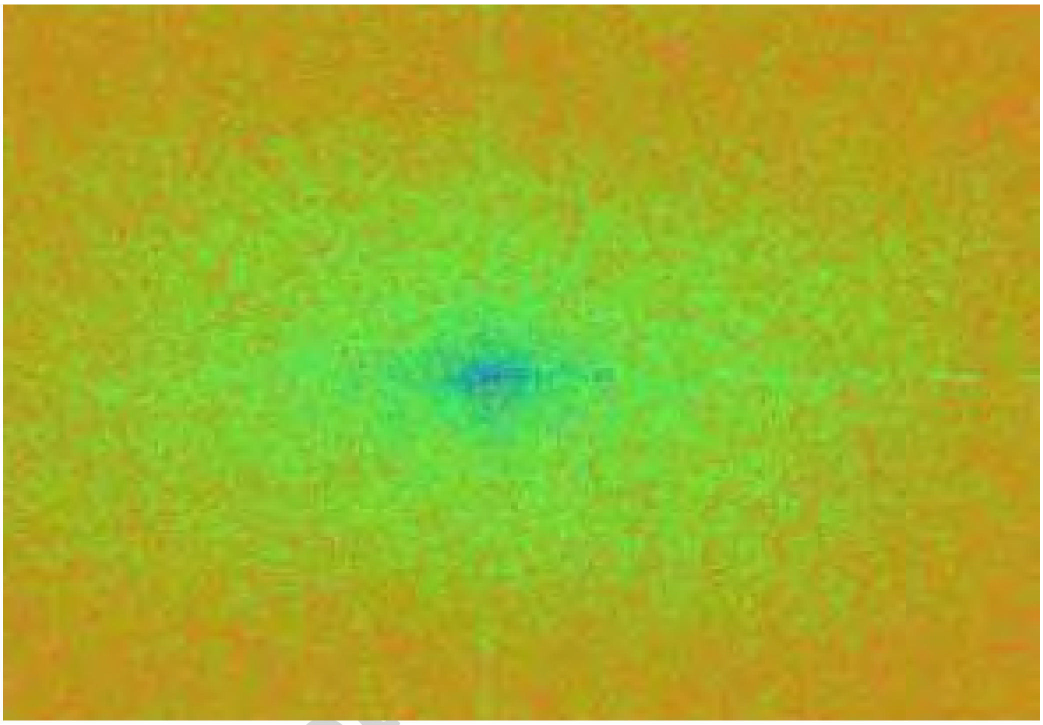


Figure 11: Fourier transform of a remotely sensed image. blue: high values. red: low values, green: medium values. The higher the green cloud the higher the magnitude of values toward the border of the image, i.e. the high frequency part. Hence the higher the green cloud the higher the heterogeneity of the image. (Please refer to the main text for additional information).

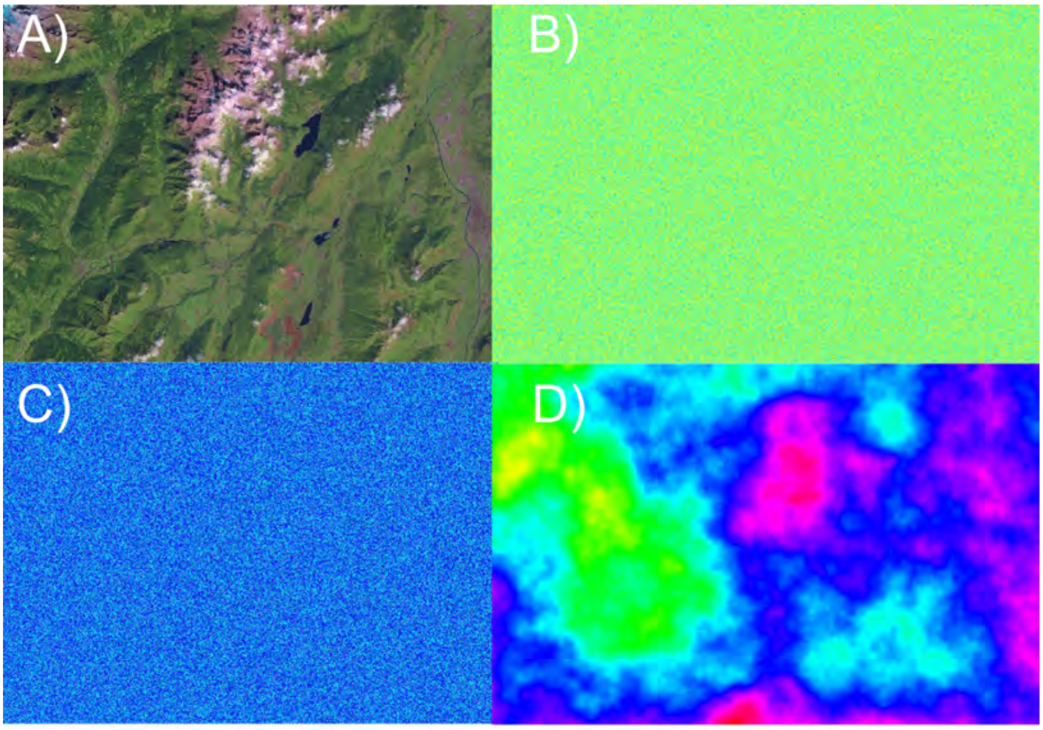


Figure 12: Random surfaces can be created as neutral models to test for patterns in real world images. As an example, patterns from a Landsat ETM+ of the Trentino region (Northern Italy) might be tested against a complete random surface (B), a gaussian surface (C), a fractal surface (D), fractal dimension 2.1.