Digital Elevation Models (DEMs) clustering for terrain modeling

E. R. Stefanescu [1], A.K. Patra[1], and M. Bursik[2]

[1]Department of Mechanical and Aerospace Engineering, University at Buffalo, Buffalo, NY 14260

[2]Department of Geology, University at Buffalo, Buffalo, NY 14260

April 17, 2013

Abstract

We consider the problem of Digital Elevation Models (DEMs) segmentation in homogeneous regions, aiming for identification of plateaux, ridges, small drainages, straight front slopes, valleys, and crests, in order to create ensembles of DEMs. These are then used in a systematic hazard analysis, resulting in a complete and complex hazard maps. In the paper we explore and implement a method for segmentation using clustering, that is required / needed when we want to construct a sparse representation of the DEM. The method – spectral clustering, is extensively and successfully used in image segmentation. It is a complex method that accounts for the spatial correlation of the elevation points and has the advantage that it can be used for almost any application where relationships between topographic features and other components of

landscapes are to be assessed. Here, the method is adapted for the case in which each data point has associated range of geomorphometric measures.

1 Introduction

Information about topography is necessary for landscape evaluation, erosion studies, hydrology and geophysical modeling, natural hazard prevention, etc. The classic way to incorporate relief units into a landscape assessment is to delineate them during field survey or using stereo aerial photographs. This approach is relatively time-consuming and the results depend on the subjective decision of the interpreter. Several methods for the creation of landform elements using elevation-derived attributes are described in the literature. Commonly, these techniques developed regions of homogeneity based on common attributes and then classified those regions (or groups of regions) as elements. The most widely used techniques are: self organizing map [Kohn et al., 1995], watershed segmentation [Najman and Schmitt, 1996], support vector machine [Gunn, 1997], segmentation using heuristic rules and fuzzy logic [Sonka et al., 1999], fuzzy K-means classification [Burrough et al., 2000] and object-based image analysis [Carleer et al., 2005]. Many of these techniques have drawbacks, especially when the method relies heavily on hydrological information and requires data-specific knowledge; also these methods don't incorporate autocorrelation between the same attribute at two locations in their models. Digital Elevation Models (DEMs) are digital representations of terrain, and are represented as an array of squared cells (data points/ pixels) with an elevation associated to each data point. They can have different resolutions (5m, 30m, 90m, 120m, etc) and can be obtained from various methods (photometry, radar interferometry, laser altimetry, etc.). Usually the size of a DEM varies from tens to hundreds of kilometers which can lead to thousands to millions of grid points.

The aim of this paper is to quantify the variation in the output of a computational flow model for block and ash flows, when the model inputs, including the elevation values represented in the DEM, are uncertain or given as a range of possible values. Integrating these variations in the possible flows as a function of input uncertainties provides well-defined data on the probability of hazard at specific locations, i.e., a hazard map [Dalbey et al., 2008]. In particular, the focus here is on assessing the influence of DEM uncertainties and propose an improved method of generating ensembles of DEMs.

In this context, we would like to implement a more complex model segmentation of a DEM of Mammoth Mountain to create non-overlapping groupings of homogeneous regions. Mammoth Mountain (Fig. 1) is a volcano located in California and it was chosen due to the ease of obtaining data sets.

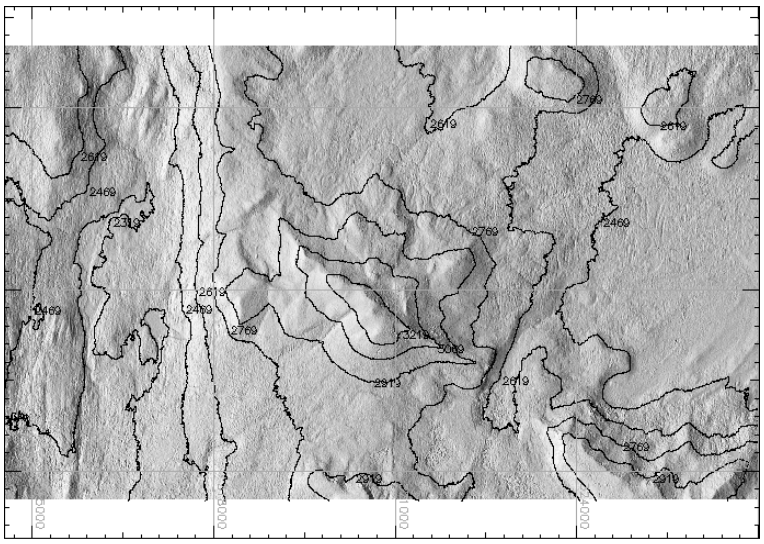


Figure 1: Hillshade plot of the Mammoth Mountain

2 Methodology

Segmentation is a broad term, covering a wide variety of problems and techniques. Segmentation methods are based on some data point or region similarity in relation to their local neighborhood. A variety of different methods have been proposed for image segmentation such as boundary-based segmentation, region-based segmentation and pixel-labeling. One view of segmentation is that we are trying to determine which components of the data set naturally "belong together". This is a problem known as *clustering*. Clustering is difficult for a number of reasons. Real-life data may contain clusters of varying size and shape, whose number is unknown in advance. Noise and outliners can further complicate the task by connecting separate clusters. Spectral clustering was firstly developed in the context of graph partitioning problems [Donath and Hofmann, 1973], where the problem is to partition a weighted graph into disjoint pieces, minimizing the sum of the weights of the edges linking the disjoint pieces. In the graph the nodes represent the grid points and arcs represent affinities ("couplings") between nearby grid points. The final cluster assignments of the dataset can be achieved by optimizing some clustering criteria defined on the graph. The criteria of some spectral clustering methods are to optimize some cut value on an undirected graph, such as normalized cut [Shi and Malik, 2000], ratio cut [Hagen and Kahng, 1992], and min–max cut [Ding et al., 2001].

To be able to perform the segmentation of the DEM in homogeneous regions we need to specify a range of geomorphometric measures which can be extracted from the surface. We define a *feature matrix* of DEM attributes, consisting of elevation and first and second derivatives of elevation (slope, profile curvature and tangential curvature). Slope and curvature are easily extracted from a DEM within a Geographical Information System (GIS).

In the next sections basic methodology for generating ensembles of DEMs using segmentation is presented, with emphasis on segmentation using spectral clustering. Subsequent

sections summarize the TITAN2D flow simulation tool and its use in a systematic hazard analysis. The hazard analysis tool itself uses ensembles of TITAN2D simulations to construct statistical surrogate models of flow outcomes at different locations as a function of model inputs, such as flow volume and pile initial location. Sampling of these surrogates leads to the construction of effective hazard maps that reflect the range of uncertainty in the model inputs.

2.1 Spectral Clustering

A digital representation of a terrain surface is an approximation of reality and is often subject to significant error. The error is usually not known in terms of both magnitude and spatial distribution. There are, in fact large uncertainties associated with the construction of DEMs. DEM vendors generally provide users with a measure of vertical accuracy in the form of the root mean squared error (RMSE) statistic. One key feature of the DEM grid points, which are spatial data, is the autocorrelation of observations in space. Generally, spatial autocorrelation refers to the correlation between the same attribute at two locations. Observations in close spatial proximity tend to be more related than observations at larger distances or separation. Based on this assumption spectral clustering is performed to identify homogenous regions within a DEM.

Compared with traditional clustering algorithms, spectral clustering has some advantages: can stably detect non-convex patterns and linearly non-separable clusters [Sakai and Imiya, 2009], and can obtain the globally optimal solutions in a continuous domain by eigendecomposition [Archip et al., 2005]. As a discriminative approach, they do not make assumptions about the global structure of data. Instead, local evidence on how likely two data points belong to the same class is first collected and a global decision is then made to divide all data points into disjunct sets according to some criterion. Often, such a criterion

can be interpreted in an embedding framework, where the grouping relationships among data points are preserved as much as possible in a lower-dimensional representation. What makes spectral methods appealing is that their global-optima in the relaxed continuous domain are obtained by eigendecomposition. However, to get a discrete solution from eigenvectors often requires solving another clustering problem, albeit in a lower-dimensional space. That is, eigenvectors are treated as geometrical coordinates of a point set. Unfortunately, when the number of grid points (denoted as n) is large, spectral clustering encounters a quadratic resource bottleneck in computing pairwise similarity between n nodes and storing that large matrix. Moreover, the algorithm requires considerable computational time to find the smallest k eigenvalues of a Laplacian matrix. Eigenvalues and eigenvectors are at the heart of spectral clustering algorithms, and in spite of their importance, existing eigensolvers do not scale well. Fast computation schemes for spectral clustering have been proposed by different authors. They focus on the eigenvector computation of a graph Laplacian defined by a matrix of data similarities. The Krylov subspace methods [Mahadevan, 2008], are iterative algorithms for finding leading eigencomponents of a sparse matrix, while Dhillon et al. [2007] assume the availability of the similarity matrix and propose a method that does not use eigenvectors. Fowlkes et al. [2004] propose using the Nyström approximation to avoid calculating the whole similarity matrix. In this paper we are using a method proposed by Song et al. [2008], which parallelize spectral clustering on distributed computers to address resource bottlenecks of both memory use and computation time.

2.1.1 Approach

For a given data set $P = \{p_1, \ldots, p_n \in R^d\}$, spectral clustering finds a set of data clusters, $\{C_1, \ldots, C_k \subset P\}$, on the basis of spectral analysis of a similarity graph. Spectral clustering builds a weighted graph $G(V, E)$, where V represents vertices and E, edges. We represent

each elevation point as a node in the graph G and the links between the adjacent data points will form the edges of the graph. Spectral clustering partitions data points into groups such that the members of a group are similar to each other and dissimilar to data points outside of the group. Given data points, an affinity matrix can be represented by a weighted adjacency matrix $W \in \mathcal{R}^{n \times n}$, where w_{ij} is a measure of the similarity between grid point i and grid point j. The affinity matrix is used to preserve the local structure of the patterns. It expresses the degree of similarity between points, and it must have the following properties: i) non-negative; ii) symmetric; iii) invertible. We have chosen the heat kernel for calculating the affinity matrix, as:

$$\mathbf{W}_{ij} = \begin{cases} \exp \frac{-\|F(i)-F(j)\|}{\sigma_F^2} * exp \frac{-\|x(i)-x(j)\|}{\sigma_x^2}, & if \parallel x(i) - x(j) \parallel \leq r \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

where $F(i)$ represents the DEM feature vector for node i, and $x(i)$ represents the coordinate location of i^{th} node. σ_F and σ_x are positive tuning parameter that controls the decay of the affinity [Tung et al., 2010]. The graph partitioning can be interpreted as a minimization problem of an objective function. Common objective functions are the ratio cut (Rcut), normalized cut (Ncut) and min–max cut (Mcut) expressed as:

$$Rcut(C_1, \ldots, C_k) = \sum_{l=1}^{k} \frac{(C_l, P \setminus C_l)}{card\ C_l} \tag{2}$$

$$NcutC_1, \ldots, C_k) = \sum_{l=1}^{k} \frac{(C_l, P \setminus C_l)}{cut(C_l, P)} \tag{3}$$

and

$$Mcut(C_1, \ldots, C_k) = \sum_{l=1}^{k} \frac{(C_l, P \setminus C_l)}{cut(C_l, C_l)} \tag{4}$$

Here, $cut(X, Y)$ is the sum of the edge weights between $\forall p \in X$ and $\forall p \in Y$, $P \setminus C_l$ is the complement of $C_l \subset P$, and $card\ C_l$ denotes the number of points in C_l. The degree d_i of

node i is the sum of all edge weights incident on x_i:

$$d_i = \sum_{j=1}^{n} w_{ij} \tag{5}$$

The *degree matrix* $D \in \mathcal{R}^{n \times n}$ is defined as the diagonal matrix with the degrees $d_1, \ldots, d_n$ on the diagonal, while the normalized graph Laplacian matrix $L \in \mathcal{R}^{n \times n}$ is defined as [Chung, 1997, Shi and Malik, 2000, Ng et al., 2002]:

$$L = I - D^{-1/2} W D^{-1/2} \tag{6}$$

The data clustering by graph-cut boils down to the eigenvalue decomposition problem of L. These eigenvectors induce an embedding of the data points in a low-dimensional subspace wherein a partitioning based on the normalized cut (NCut) can be used. The solution of the minimization problem can be obtained from the Fiedler eigenvector [Yu and Shi, 2003]. The steps involved in DEM segmentation using spectral clustering are summarized in Figure 2.

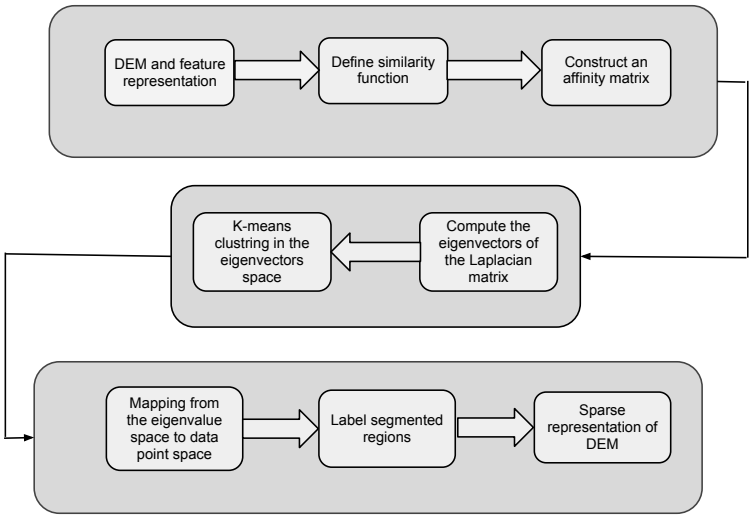


Figure 2: Spectral clustering for DEM segmentation workflow

2.1.2 Parallel implementation

When the number of data points (n) is large, the computational complexity of spectral decompositions can reach $O(n^3)$ (W dense). If the affinity matrix W is define as in Equation 1 then its construction takes $\mathcal{O}(n^2 d)$ flops, which can be also computationally intensive if the data cardinality n or dimensionality d is large. Chen et al. [2011] investigate approaches for large-scale spectral clustering and propose a parallel implementation, which is also adopted in this paper. The strategy to address the computational and memory difficulties involves distributing n data instances onto p distributed machine node. On each node, parallel spectral clustering (PSC) computes the similarities between local data and the whole set in a way that uses minimal disk I/O. Then PSC stores the eigenvector matrix on distributed nodes to reduce per-node memory use. Together with parallel eigensolver and K-means, PSC archives good speedup with large data sets.

3 DEM ensembles

When propagating uncertainty in DEMs through a geophysical system, stochastic methods are considered to be an effective way to estimate the probability density function of outputs by addressing uncertainties present in initial conditions and in model approximations. In previous work Stefanescu et al. [2012] presented a basic methodology for generating ensembles of DEMs representative of the true DEM. Here, we are extended the methodology at the cluster level, by making the assumption that each homogenous regions has its own error model which leads to different random fields. The random fields are used in creating multiple equally likely representations of an actual terrain surface, following the approached suggested by Ehlschlaeger and Goodchild [1994]. A normal distribution (mean of 0.0 and variance of 1.0) of maps or realizations is computed to reproduce the spatial autocorrelation encountered

in the original error surface, filtered using a Gaussian convolution filter, with kernel sizes derived from autocorrelation analysis of the original error surfaces. The random field function derives its spatial dependence from the use of a distance based decay filter function. The following equation is used to generate the random field:

$$Z(\mathcal{U}) = \frac{\sum_v w_{u,v}\epsilon_v}{\sqrt{\sum_v w_{u,v}^2}}, \quad u \in \mathcal{U},\ v \in \mathcal{V} \tag{7}$$

$$w_{u,v} = \begin{cases} 1 & : d_{u,v} \leq F \\ \left(1 - \frac{d_{u,v}-F}{D-F}\right)^E & F < d_{u,v} \leq D,\ u \in \mathcal{U},\ v \in \mathcal{V} \\ 0 & : d_{u,v} > D \end{cases} \tag{8}$$

where $\mathcal{V}$ is the set of points potentially influencing points in a given area, $\mathcal{U}$, $w_{u,v}$ is the spatial autocorrelative effect between points $u \in \mathcal{U}$ and $v \in \mathcal{V}$, ϵ_v is a Gaussian random variable with a mean of 0 and variance of 1, $d_{u,v}$ is the distance between u and v, D is the minimum distance of spatial independence, E is the distance decay exponent, and F the distance at which errors are completely correlated.

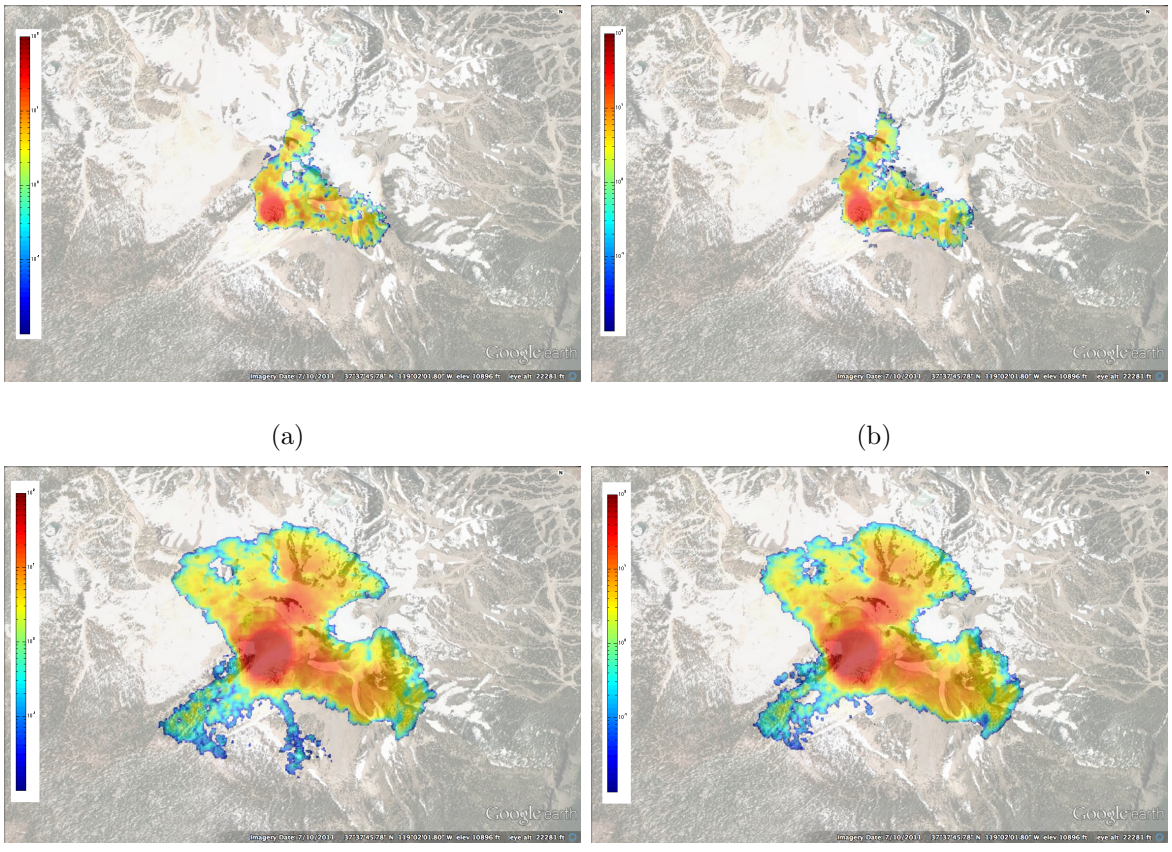
A set of random fields are created for each homogenous region/ cluster and are calibrated to the spatial variation of the field being simulated using a correlogram function. This is done by fitting the correlogram and choosing the best descriptive parameters of the random field (the minimum distance of spatial independence, the correlated distance decay exponent and the filter parameter) in a weighted least-square estimator. After running hundreds of tests with multiple combinations of D, E and F, the best random field was found by fitting the error map characteristics such that the sum of least squares difference between an error field's correlogram and the target correlogram is minimized.

Each error realization was added to the "true" DEM indicated as $m(\mathcal{U})$, to generate equally probable realizations of the topography for the error structure of a DEM under

consideration:

$$R(\mathcal{U}) = m(\mathcal{U}) + m(m(\mathcal{T})) + (m(s^2(\mathcal{T})) \cdot \epsilon) \cdot Z(\mathcal{U}) \tag{9}$$

where $R(\mathcal{U})$ is a realization of the elevation dataset $m(\mathcal{U})$, $\mathcal{T}$ is a group of sets of spatially uncorrelated sample points in $m(\mathcal{U})$, and ϵ is a Gaussian random variable with mean 0.0 and variance 1.0. $m(m(\mathcal{T}))$ and variance $m(s^2(\mathcal{T})$ is mean and variance, respectively, of all sets in $\mathcal{T}$. $Z(\mathcal{U})$ specifies the random field as defined in Equation 7 for each homogenous region.



(a) (b)

(c) (d)

Figure 3: Flow maps for two selected parameters values when DEM was created using a) no cluster b) cluster

4 Flow simulator and Hazard map construction

5 Results and Conclusions

5.1 Earth Mover's Distance

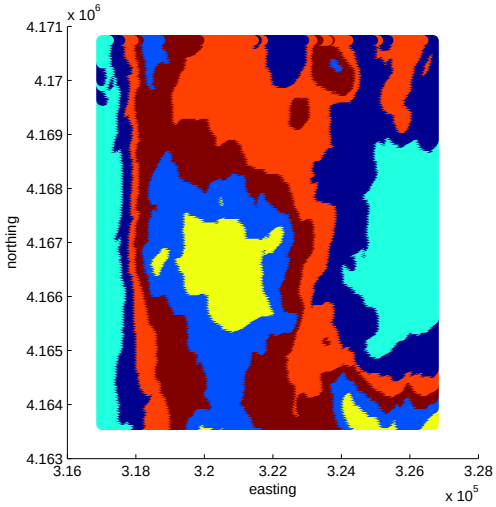
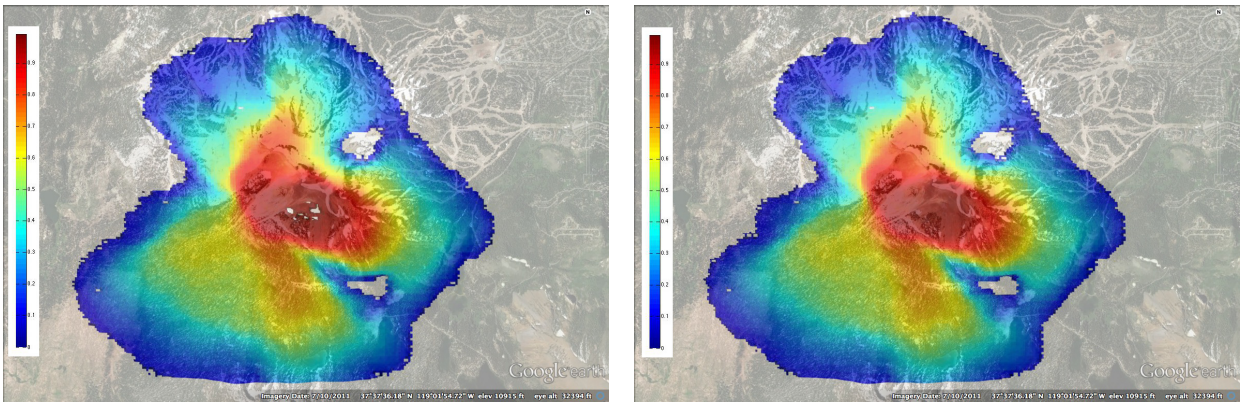


Figure 4: Parallel spectral clustering



(a) (b)

Figure 5: Hazard map a) no cluster b)clusters

Acknowledgment

References

N. Archip, P. Rohling, H. Tahmasebpour, and S.K. Warfield. Spectral clustering algorithms for ultrasound image segmentation. *in: MICCAI 2005, Lecture notes in Computer Science*, 3750:862–869, 2005.

P.A. Burrough, P.F.M. van Gaans, and R.A. McMillan. High-resolution landform classification using fuzzy k-means. *Fuzzy Sets and Systems*, 113:37–52, 2000.

A.P. Carleer, O. Debeir, and E. Wolff. Assessment of very high spatial resolution satellite image segmentations. *Photogrammetric Engineering and Remote Sensing*, pages 1285–1294, 2005.

W.Y. Chen, Y. Song, H. Bai, C.J. Lin, and E.Y. Chang. Parallel spectral clustering in distributed systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33 (3):568–586, 2011.

F.R.K Chung. Spectral graph theory. *CBMS Regional Conference Series in Mathematics, American Mathematical Society*, 92, 1997.

K. Dalbey, A.K. Patra, E.B. Pitman, M.I. Bursik, and M.F. Sheridan. Input uncertainty propagation methods and hazard mapping of geophysical mass flow. *Journal of Geophysical Research*, 113:5203–5219, 2008. doi:10.1029/2006JB004471.

I.S. Dhillon, P. Guan, and B. Kulis. Weighted graph cuts without eigenvectors: A multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29 (11): 1944–1957, 2007.

C.H.Q Ding, X. He, M. Gu, and H.D. Simon. A minmax cut algorithm for graph partitioning and data clustering. *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 107–114, 2001.

W.E. Donath and A.J. Hofmann. Lower bounds for the partitioning of graphs. *IBM Journal of Research and Development*, 17:420–425, 1973.

C. Ehlschlaeger and M.F. Goodchild. Uncertainty in spatial data: Defining, visualizing, and managing data errors. In *Proceedings GIS/LIS'94*, pages 246–253, 1994.

C. Fowlkes, F. Belongie, F. Chung, and J. Malik. Spectral grouping using the nystrom method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26 (2):214–225, 2004.

S.R. Gunn. Support vector machines for classification and regression. *Technical report image speech and intelligent group Southampton University*, 1997.

L. Hagen and A. Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11 (9): 1074–1085, 1992.

J. Kohn, M.S. Suk, and S.M. Bhandarkar. A multilayer self organizing feature map for range image segmentation. *Neural Networks*, 8 (1), 1995.

S. Mahadevan. Fast spectral learning using Lanczos eigenspace projections. *in AAAI*, pages 1472–1475, 2008.

H. Mitasova, J.Hofierka, M. Zlocha, and L.R. Iverson. Modeling topographic potential for errosion and deposition using GIS. *International Journal of Geographical Information Systems*, 10:629–641, 1996.

L. Najman and M. Schmitt. Geodesic saliency of watershed contours and hierarchical segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12): 1163–1173, 1996.

A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. *In T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems*, 14:849–856, 2002.

T. Sakai and A. Imiya. Fast spectral clustering with random projection and sampling. *Machine Learning and Data Mining in Pattern Recognition*, pages 372–384, 2009.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (8):888–905, 2000.

Y. Song, W.Y. Chen, H. Bai, C.J. Lin, and E.Y. Chang. Parallel spectral clustering. *in: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part II*, 5215:374–389, 2008.

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

E.R. Stefanescu, M. Bursik, G. Cordoba, K. Dalbey, M.D. Jones, A.K. Patra, D.C. Pieri, E.B. Pitman, and M.F. Sheridan. Digital Elevation Model (DEM) uncertainty and hazard analysis using a geophysical flow mode. *Proceedings of Royal Society A*, 2012. in press.

F. Tung, A. Wong, and D.A. Clausi. Enabling scalable spectral clustering for image segmentation. *Pattern Recognition*, 43:4069–4076, 2010.

S.X. Yu and J. Shi. Multiclass spectral clustering. *In: Proc. Internat. Conf. on Computer Vision*, pages 313–319, 2003.