A Continuous scale-space method for the automated placement of spot heights on maps

Luigi Rocca[a,*], Bernhard Jenny[b,1], Enrico Puppo[a]

[a] *Department of Informatics, Bioengineering, Robotics and System Engineering (DIBRIS), University of Genova, Italy*
[b] *College of Earth, Ocean, and Atmospheric Sciences, Oregon State University, Corvallis, OR, USA*

Abstract

Spot heights and soundings explicitly indicate terrain elevation on cartographic maps. Cartographers have developed design principles for the manual selection, placement, labeling, and generalization of spot height locations, but these processes are work-intensive and expensive. Finding an algorithmic criterion that matches the cartographers' judgment in ranking the significance of features on a terrain is a difficult endeavor. This article proposes a method for the automated selection of spot heights locations representing natural features such as peaks, saddles and depressions. A lifespan of critical points in a continuous scale-space model is employed as the main measure of the importance of features, and an algorithm and a data structure for its computation are described. We also introduce a method for the comparison of algorithmically computed spot height locations with manually produced reference compilations. The new method is compared with two known techniques from the literature. Results show spot height locations that are closer to reference spot heights produced manually by swisstopo cartographers, compared to previous techniques. The introduced method can be applied to elevation models for the creation of topographic and bathymetric maps. It also ranks the importance of extracted spot height locations, which allows for a variation in the size of symbols and labels according to the significance of represented features. The importance ranking could also be useful for adjusting spot height density of zoomable maps in real time.

Keywords: scale-space, spot heights, soundings, map generalization, terrain mapping

1. Introduction: placing spot heights on maps

Spot heights are included in topographic maps to quickly and accurately ascertain the elevation values of points on a terrain surface. For locations below sea level, spot heights are referred to as depth points or soundings. For important summits and mountain passes, the name of the mountain or pass is commonly placed next to the spot height symbol in addition to the elevation. To indicate the relative importance of topographic features, the type size of toponyms is graded so that larger type indicates more important features. Additionally, roman and italic type can be used to differentiate between mountain peaks and other points. Cartographers carefully adjust the density and distribution of spot heights to the scale and purpose of the map and the type of terrain represented. A rugged mountain region with numerous summits and passes requires more of them than a flat area (Imhof, 2007).

Ideal locations for spot heights are sites that, when marked on the map, are unambiguous and easy to associate with visible features and landmarks. Examples include natural features, such as mountain passes, summits, the lowest points in depressions, junctions of streams, edges of terraces, and artificial landmarks, such as intersections in road networks, churches, railway stations, bridges, isolated buildings and tunnel portals (Imhof, 2007). Spot heights are not as useful when the location of the point cannot be identified accurately, such as in the middle of a slope or a flat plane or on a flat hillcrest (Gilgen, 2014), as depicted in Figure 1. The selection, placement, labeling, and generalization of spot heights are time consuming and considerably expensive when done manually (Baella and Pla, 1999; Baella et al., 2007). Therefore, accelerating the process for generating spot heights has clear economic advantages for the production of large-scale topographic maps.

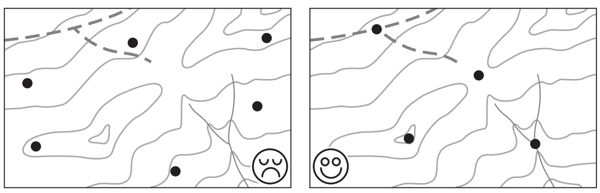


Figure 1: Poor and good placement of spot heights (after Spiess, 1996).

*Corresponding author
Email address: rocca@dibris.unige.it (Luigi Rocca)
[1]Present address: School of Science, Geospatial Science, RMIT University, Melbourne, Australia

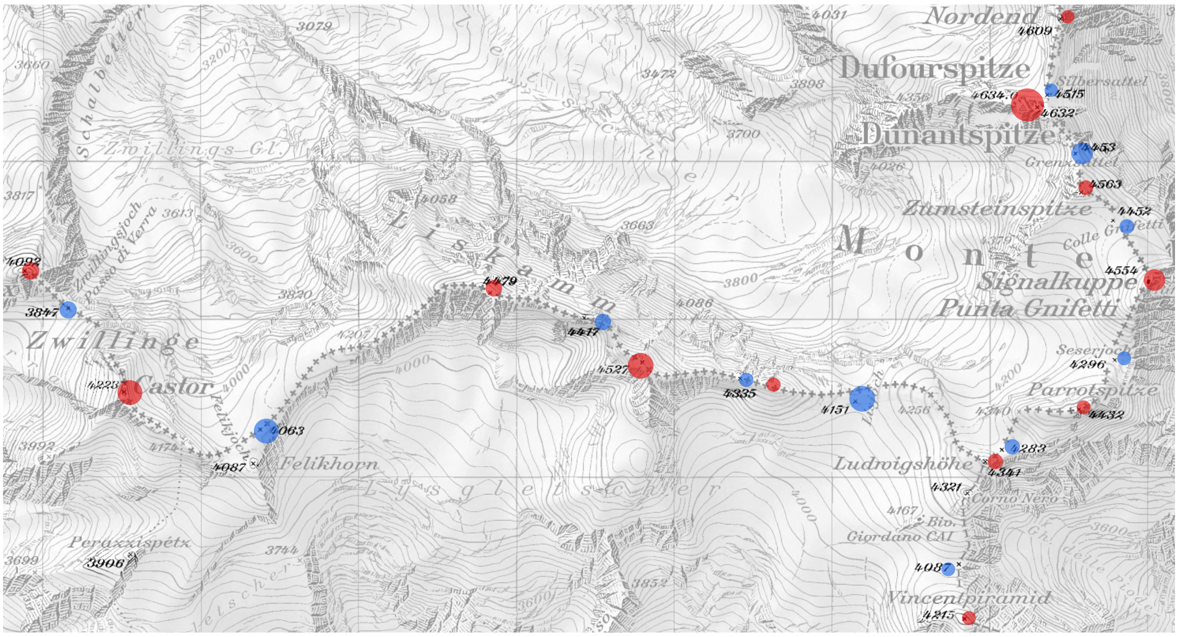


Figure 2: Spot heights locations selected by the proposed method; red: summits; blue: mountain passes. Circle size indicates importance values. Background: 1:50,000 map by swisstopo. In the background map, reference spot heights representing peaks and saddles are highlighted in black. The selection has been obtained using different lifespan parameters: 5.32 for mountain passes and 4.51 for summits. Almost all spot heights representing summits and passes in the map have been successfully matched, with only one false positive present, and their automatically computed importance adequately reflects their actual significance on the terrain.

The aim of our work is to offer an algorithmic method for the automated placement and selection of spot height locations representing natural features, as extracted from a digital elevation model (Figure 2). In particular, we focus on mountain peaks, saddles and depressions. Techniques for the extraction of points representing morphological features in digital terrains have been known for a very long time (Peucker and Douglas, 1975). A simplistic extraction method, though, is not enough, because the points that would make good spot heights will be just a small subset, lost among countless others representing noise or unimportant features, especially when using a detailed model. A trivial solution would be to apply a smoothing filter in order to clean the terrain of noise and high frequency information; this, however, would displace the position of surviving features, making them useless for our purpose unless manually corrected. An optimal solution is represented by a ranking of all extracted features, as found in the original source, according to their intrinsic importance. This allows for an easy discrimination between features to be retained and to be discarded, according to the desired amount of generalization and depending on the type and scale of the designed map. It also permits differentiating important features from lesser ones using different symbols or font sizes.

We argue that measuring how long a feature survives in a scale-space of the original surface is a good proxy of the feature's morphological significance, provided that the computation is precise and error-free. A virtually continuous scale-space data structure is adopted as the foundation of our selection method. Such a structure provides an encoding of all morphological features extracted from the input surface and endows them with an importance score, fit to be used as a simple selection parameter by cartographers to choose the right amount of spot heights. The placement of features is as precise as the elevation model allows. Moreover, we propose a general and statistically sound method for testing how well an automatic algorithm for spot heights selection works matched to a preexisting reference. We test our results against manually compiled spot height locations representing peaks in the 1:25,000 map series by the Swiss Federal Office of Topography swisstopo, and we compare to other known automatic techniques for the selection of mountain peaks.

2. Related work

2.1. Extraction of spot height locations

The automated selection and generalization of spot heights for topographic maps has received less attention than other terrain mapping techniques (Guilbert et al., 2014). Two articles by Baella et al. (2007) and Palomar-Vázquez and Pardo-Pascual (2008) introduced a method for automatically selecting relevant spot height locations for small-scale maps from a database. Input locations are rated by criteria to define an importance value for each point, and then points are selected according to their importance value. Their method uses many criteria, among them the geomorphological importance of peaks, saddles, and depressions. The importance of peaks is computed by placing a circle on each point and increasing the radius until each circle contains one point higher than its center. The significance value is determined from the radius and the averaged slope of the terrain around the point. Map space is partitioned into half planes using a binary tree. In each half plane an equal number of spot heights exist, and spot heights with the lowest importance values are removed. The method is complicated and results are influenced by a large number of parameters.

Baella et al. (2007) and Palomar-Vázquez and Pardo-Pascual (2008) note that their results differ from those obtained with manual generalization. The discrepancy is attributed to the difficulty of introducing all required contextual and aesthetical rules applied in manual spot height generalization to an automated method.

Jaara and Lecordix (2011) developed a method for extracting saddles from grids. For each cell, in a grid, the neighborhood is analyzed, and a cell is considered to belong to a saddle when two neighboring regions are lower and two are higher. Because the algorithm identifies multiple neighboring cells as saddles, a local search algorithm is used to find the exact position of each saddle, discarding the spurious neighbors. They also introduce a peak detection method for placing spot heights on maps: the terrain is divided into regular squares, and for each square, the cell with the maximum altitude is found. If the altitude is higher than the highest points in the neighboring squares, the point is designated a peak. The procedure is repeated for growing square sizes. As square size grows, peaks are gradually ranked for their importance. The method is algorithmically simple and fast, but the fixed size of the square structure cannot adequately extract peaks that vary in density.

Wood (2004) introduced an efficient algorithm using the relative drop selection criteria to identify significant peaks. The raster-based method searches local peaks in the terrain model and identifies the area enclosing each peak that is bounded by an isoline, such that no other point that is higher than the peak is contained within the isoline. The difference between the peak's elevation and the elevation of the enclosing isoline is the relative drop distance. Peaks are selected by choosing a minimum relative drop distance. Wood's method successfully identifies major peaks in mountainous areas, but it may also identify irrelevant minor peaks in flat areas.

Deng and Wilson (2008) presented a multi-scale method for extracting peaks from an elevation model. The method also assigns a relevance score to each peak. They propose four criteria that characterize peaks: high slope relief in the surrounding local area; amount of steepness in that same area; high elevation compared to a larger neighborhood; and a small number of competing peaks in that same neighborhood. Those properties are quantified, normalized, and weighted together as a single score, which is computed at three scales of increasing window size and then normalized again in a final score. The proposed algorithm has the potential to be fine-tuned for different purposes and applications, but it is quite complex to configure, with several different weights to assign. It is also expensive to run—execution times up to ten hours are reported for a DEM composed by 1740×930 cells.

2.2. Morse theory

Morse theory (Milnor, 1963) links the critical points of a smooth real-valued function f to the topology of the manifold on which f is defined by studying its gradient flow. The theory extends and generalizes Maxwell's intuitions about the partition of a landscape into valleys and mountains (Maxwell, 1870). In Morse theory, areas of uniform gradient flow are part of the same region. Basins contain a minimum and are bounded by ridges, with flow lines that have the same destination. Mountains contain a maximum and are bounded by river valleys, with flow lines that have the same origin. The intersection of basins and mountains creates a partition called *Morse-Smale complex*, which has boundaries that are always divided in two by saddles. The Poincaré-Hopf index theorem states an invariant relation between the number of minima, maxima and saddles on a surface:

$$n_{maxima} + n_{minima} - n_{saddles} = 2 \tag{1}$$

A survey and evaluation of known discrete algorithms and data structures derived from Morse theory can be found in (Comic et al., 2014). Methods based on these techniques have also been widely studied in geographical information science, where many data structures conceptually equivalent to Morse-Smale complexes such as surface networks, Reeb graphs and contour trees are routinely used. These structures compactly encode a great amount of information about the underlying surface by storing constitutive landform elements (peaks, pits and saddles) and their connections by ridge and valley lines (Takahashi, 2006). A terse graph of the surface topology makes it possible to build algorithms that analyze and take into account the logical and morphological structure of the surface (Wolf, 2006). Most relevant to our purposes, these graphs can be weighted by assigning a measure of importance to their nodes and they can be simplified in a topologically sound way. Edelsbrunner et al. (2001) have proposed one of the most successful weighting and simplification techniques, known as homological persistence. Their method uses amplitude information to find a pair of features to be simplified; it then progressively simplifies the entire graph, simultaneously scoring all features according to their importance.

Likewise, the method we propose weighs surface features using a scale-space based technique (Rocca and Puppo, 2013), which is guided by frequency information: it starts by identifying critical points on the surface and it takes into account the relationship described in Equation 1. The working paper on surface networks by Rana and Morley (2002) reports several questions that are still open regarding techniques based on Morse theory and their geographic applications. Chief among them is the question of scale, and its relationship with any chosen feature detection method: on one hand, discrete methods find features at the highest possible frequency; on the other hand, kernel-based methods often cause loss of the correct topological relations. It is our hope that the method we propose can bridge a gap between these two different classes of approaches.

2.3. Scale-space techniques for feature tracking

Since its introduction by Witkin (1983) and Koenderink (1984), scale-space methods have been widely studied in the computer vision and image processing fields (see Price, 2012, s. 4.4). A standard scale-space consists of a discrete collection of subsequently filtered versions of an input signal f, usually obtained by repeated convolution of f with a Gaussian kernel. Each level of the scale-space emphasizes features at different scales, from finer to coarser. These levels can be seen as discrete samples of a diffusion process that has, in principle, a continuous nature; an example of a scale-space of a digital terrain model is shown in Figure 3.

The *deep structure* of scale-space (Lindeberg, 1994) tracks the zero-crossings of the signal's differential invariants in consecutive levels of the scale-space, thereby capturing the evolution of features across scales. In a scalar field, these features correspond to local extrema, or *critical points*; in the special case of a terrain model they directly represent peaks, pits, and saddles on the surface.

The relative importance of critical points in the original signal is computed by measuring how long their life span is in the scale-space, as more important features survive smoothing while less important ones disappear. When critical points disappear, smoothed out by the filter, they always do so in maximum-saddle and minimum-saddle pairs, in accordance to Equation 1. Often, critical points that do not disappear move on the surface, displaced by the smoothing effect along the direction of diffusion. The axiomatic linear scale-space theory (Lindeberg, 2011) provides many different guarantees, among them the principles of non-creation and non-enhancement of local extrema.

In the standard approach to scale-space, features are extracted in each level and eventually matched to a feature in the next one by local neighborhood search, starting from the feature position in the previous level. This approach tends to be error prone, with false and missed matches, because a lot of features disappear from one level to the next, and the relative movement of a feature between levels can be wide. Moreover, the tracking of features across levels is difficult and inherently unreliable because the principle of non-creation of local-extrema is not valid when smoothing multi-dimensional signals: Gingold and Zorin (2007) observe that the number of features sometimes increases, with new features always appearing in pairs—a saddle coupled to either a maximum or a minimum.

Reininghaus et al. (2011) propose a tracking algorithm that follows the gradient of the scalar field when detecting the correspondence of critical points between consecutive levels of the scale-space. This method is more robust than standard neighborhood search but cannot overcome the many drawbacks of the discrete approach, because the granularity of the performed analysis still corresponds to the number of smoothed samples in the scale-space.

Rocca and Puppo (2013) describe a method that, starting from the discrete samples of a traditional scale-space, uses linear interpolation to obtain a virtually continuous approximation of the diffusion process. They achieve a measure of the importance of critical points that is several order of magnitudes finer than the number of levels in the input scale-space. Their experiments show that the appearance of new pairs is more common than previously thought and that a subset of newborn features can achieve very long life spans in the scale-space, sometimes taking the place of important features on the original terrain. This makes a correct measurement of the importance of critical points on the original surface impossible, because somewhere in the scale-space a tracking path that should have been longer

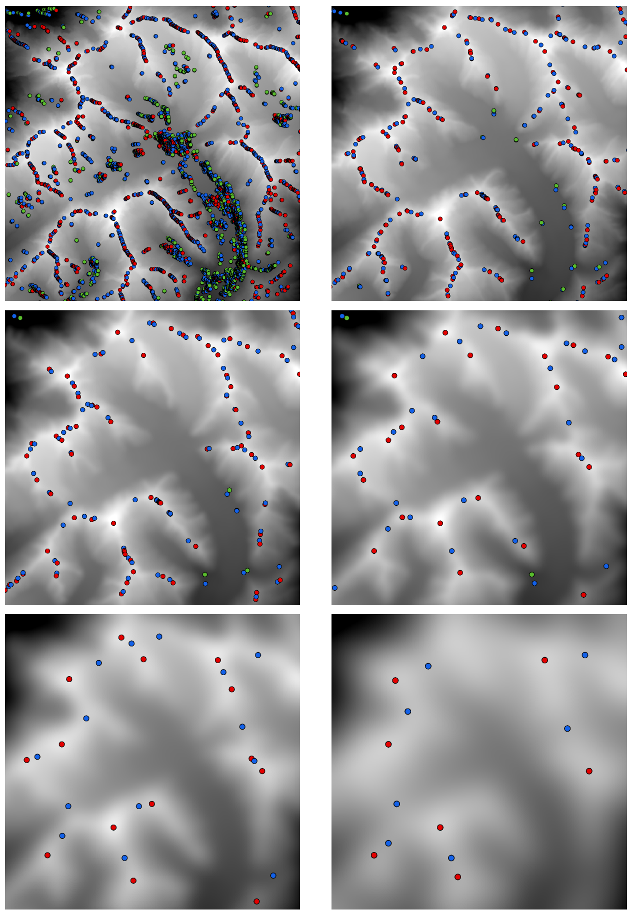


Figure 3: A discrete scale-space of a digital terrain (Aletsch Glacier region). The original surface is progressively smoothed to identify features at different scales, and the number of critical points correspondingly decreases (red: maxima; green: minima; blue: saddles).

has been interrupted by a sequence of death and birth events. This is an open problem in current literature.

3. Selection method

The method we propose for selecting spot height locations is based on multiple contributions. We suggest adoption of the life span of critical points in a linear scale-space as the main selection parameter, because the life span of a feature directly measures its morphological importance. We call this the *lifespan* parameter. Section 3.1 describes an algorithm and a data structure first introduced by Rocca and Puppo (2013), which we adapt and tailor to the specific task of computing the life span of critical points in the scale-space of a digital elevation model. At this stage, however, the result is impaired by the presence of spurious features born during the smoothing process. We develop a novel post-processing enhancement algorithm, described in Section 3.2, that solves the issue of newborn features and successfully recovers the full life span of critical points. In addition, we adopt the relative drop method proposed by Wood (2004) as the source of a secondary importance measure when selecting peaks.We call this the *drop* parameter.

3.1. Virtually continuous scale-space

Our input is a grid encoding a digital elevation model. A discrete scale-space is computed by smoothing repeatedly the input, until N levels are generated. Each n_{th} level is convoluted by a discrete Gaussian filter to compute the $(n+1)_{th}$ level, using a square window of size k_n. The window size and the kernel's standard deviation have a fixed relationship, $k_n = 6\sigma_n$. We aim at a constant ratio of disappearances of features between each pair of consecutive levels in the discrete scale space. We have empirically determined that this result can be achieved by exponentially increasing the variance of the kernel through the different levels, roughly $\sigma_n^2 \approx 2^n$, which corresponds to:

$$k_n = \text{floor}\,(\sqrt{2^n} * 6) \tag{2}$$

with k incremented by one if the result is even.

Connectivity model of the input grids. In order to fulfill the Euler-Poincaré equation 1, for each level in the scale-space we consider each input pixel to be a vertex in a triangle mesh. In our experiments, every vertex is connected by edges to six neighbor vertices in a regular fashion. Each input cell at coordinates (x, y) becomes a vertex connected by edges to six neighbors at coordinates $(x-1, y-1)$, $(x-1, y)$, $(x, y+1)$, $(x+1, y+1)$, $(x+1, y)$, $(x, y-1)$. Any other triangulation strategy could be employed, such as the one described by Rana and Morley (2002). Our choice greatly simplifies computations because of the regular structure of the mesh. Vertices on the border are connected to a virtual global minimum set to $-\infty$, thus making our mesh borderless, with every vertex except the outer one having six neighbors (Figure 4). It is assumed that two connected vertices never have the same value; one has to be higher or lower than the other. When flat areas are present a removal strategy must be employed (see Magillo et al., 2013). In our experiments, a disambiguation of the original signal is enough. If the input does not contain flat areas to begin with, flat areas do not appear in the smoothed levels of the scale-space.

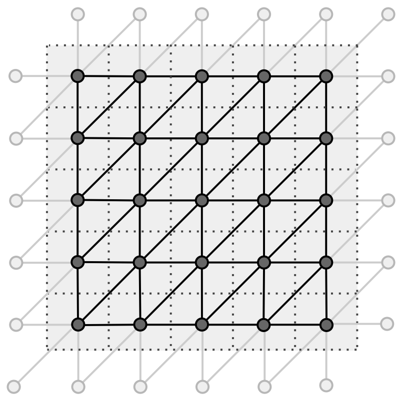


Figure 4: From the input grid (dashed lines and light grey background) to a triangle mesh (dark grey solid lines and points). Vertices of the triangle mesh are shown in the center of each input cell. Border vertices, which would have only two, three, or four edges depending on their position, are connected to a virtual outer border (light grey solid lines and points). Vertices in the virtual outer border hold a large negative value that represents a global minimum.

For each level of the scale-space, vertices are classified as sloped or critical points by analyzing their local neighborhoods, as commonly used (Figure 5). The main criterion is the number of times that the surface surrounding the central vertex goes from higher to lower. The result is always an even number between zero and six. The current vertex is then classified as:

- a *sloped* point, if the sign changes two times;
- a *maximum*, if the sign never changes and the central vertex is highest;
- a *minimum*, if the sign never changes and the central vertex is lowest;
- a *saddle*, if the sign changes four times;
- a *monkey saddle*, if the sign changes six times.

Sloped points are discarded. For each level, a list of critical points composed by peaks, pits, and saddles is retained. To guarantee consistency with Equation 1, our model treats monkey saddles as two different saddles occurring in the same spot.

Approximation of the diffusion process by linear interpolation. Levels of the scale-space, generated by Gaussian filtering, represent discrete samples of a continuous diffusion process. This smoothing process can be seen as a time-dependent dimension, starting at time t_0 (the original, unfiltered data), and regularly sampled at intervals $(t_1, t_2, \cdots, t_{N-2}, t_{N-1})$, for the N levels of the scale-space. Height values of vertices in the DEM therefore change in time, and their values across the time dimension can be linearly interpolated from the discrete samples. The resulting model is depicted in Figure 6.

Our goal is to compute how the set of critical points evolves from points extracted in the original DEM to points present in the last level of the scale-space. Our strategy is to identify all the single events that could cause a critical point to change. Given that vertices are classified based on the relation (lower or higher) they share with their neighbors, one of those relations has to change for the resulting classification to change. This happens only if two connected vertices change their relative height from one level to the next in the scale-space. This event is called *edge flip* (Rocca and Puppo, 2013). An edge flip occurs if, given two adjacent vertices a and b, with height values a_i, a_{i+1} and b_i, b_{i+1} in two consecutive levels respectively, the following condition is true:

$$(a_i > b_i \wedge a_{i+1} < b_{i+1}) \vee (a_i < b_i \wedge a_{i+1} > b_{i+1}) \qquad (3)$$

For each flip we compute a timestamp t_{flip} by interpolating the moment when the two vertices have equal height between consecutive levels:

$$t_{flip} = \frac{b_i - a_i}{b_i - a_i + a_{i+1} - b_{i+1}} + i \qquad (4)$$

Note that $i < t_{flip} < i+1$, where i and $i+1$ are the indices of two consecutive levels. For each level in the scale-space every edge in a level is compared to the corresponding edge in the next level. A flip is represented by an edge and a timestamp. All flips are collected in an array, which is then sorted according to timestamps.

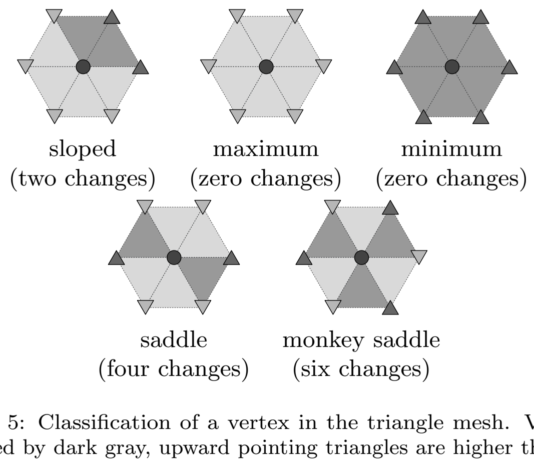


Figure 5: Classification of a vertex in the triangle mesh. Vertices depicted by dark gray, upward pointing triangles are higher than the central vertex; vertices depicted by light gray, downward pointing triangles are lower than the central vertex. The six neighbor vertices are visited in a clockwise or counterclockwise radial order, and their height values are compared to the central vertex. The number of times the sign of the comparison changes, because the relation goes from lower to higher or from higher to lower, is counted and the vertex is classified accordingly.

Tracking of features. Tracking is performed by processing the list of flips, sorted by timestamp, as described in Rocca and Puppo (2013). The algorithm's input is the list of critical points in the first level of the scale-space, and the ordered list of flips. It returns the list of critical points, modified by processing the list of flips. It makes use of a temporary map (a grid of the same dimensions as the original terrain) which keeps, for each vertex, the following information:

- The relation of the vertex with its six neighbors (lower or higher);
- Up to two pointers to critical points in the tracking data structure (none if no critical point is present; one for a peak, pit, or saddle; two if a monkey saddle occurs in the given vertex). A pointer can be implemented as an index in the list of critical points.

The temporary map is initialized with the relations present in the first level of the scale-space and the corresponding critical points. Flips are visited in order. For each flip, we look at the relations stored in the temporary map for both vertices a and b involved in the event. This allows us to classify the vertices before and after the flip. If the classification changes, one of the following events has happened:

- A critical point has **moved**. The pointer in the temporary map is erased in the source vertex and written in the destination vertex.
- A pair of critical points has **collapsed**. The two pointers are erased from the temporary map, and the two critical points involved are updated by memorizing the timestamp of their death and a pointer to their death companion.
- A pair of critical points has **appeared**. Two new critical points are added to the list, with a pointer to their birth companion, and their pointers are written in the temporary map.

Displacements	**Collapses**	**Appearances**
$(m, r) \to (r, m)$	$(m, s) \to (r, r)$	$(r, r) \to (m, s)$
$(M, r) \to (r, M)$	$(M, s) \to (r, r)$	$(r, r) \to (M, s)$
$(s, r) \to (r, s)$	$(m, K) \to (r, s)$	$(r, s) \to (m, K)$
$(K, r) \to (r, K)$	$(M, K) \to (r, s)$	$(r, s) \to (M, K)$
$(K, r) \to (s, s)$		
$(s, s) \to (K, r)$		
$(K, s) \to (s, K)$		

Table 1: Possible transitions in the state of a pair of vertices connected by a flipping edge, after Rocca and Puppo (2013). r: a sloped point; M: a maximum; m: a minimum; s: a saddle; K: a monkey saddle. Note that for every event a specular one is also possible, for a total of 32 possible events. An example would be: $(M, K) \to (r, s)$ is equivalent to $(K, M) \to (s, r)$.

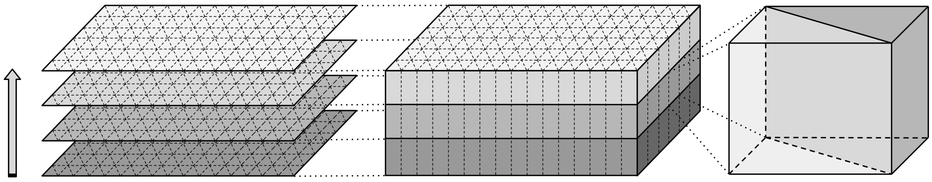


Figure 6: Abstract model of a virtually continuous scale-space. The smoothing process proceeds from the first level at the bottom, representing the original DEM (dark gray), to the last level at the top, representing the most smoothed sample (light gray). A number of levels between 10 and 20, depending on the dimensions of the input grid, worked well in our experiments. Starting from the discrete samples, organized in a triangle mesh, a continuous approximation is built using linear interpolation along both the spatial dimension and the scale dimension.

The temporary map is updated by inverting the relations of the two vertices involved in the event. A comprehensive list of the possible events is outlined in Table 1.

When all flips have been processed, critical points that have survived up to the last level get t_{N-1} as a timestamp. Each critical point in the final list has the following properties: a *timestamp* value corresponding to the moment of its death in the scale-space; an optional pointer *death-mate* that refers to the critical point this point died with, if such an event happened; and an optional pointer *birth-mate* that refers to the critical point this point was born with (if the point is not among the original ones).

3.2. Life span computation

The goal is to compute the life span in the scale-space of critical points present in the original, unfiltered terrain. The timestamp computed in Section 3.1 represent only a partial account, because of the additional, spurious points caused by the smoothing process. The birth of new critical points represents a perturbation in the usual flow of displacements and collapses, which should result in a steadily decreasing number of critical points as the scale parameter increases. Most newborn critical points are ephemeral, existing only for a very small fraction of the scale-space before collapsing again, and can be safely discarded. However, a small but sizable fraction of them does not disappear and survives arbitrarily long through scales. It turns out that, in most cases, these long-lived newborn critical points in fact extend the life span of pre-existing critical points of the same type, which disappear shortly after the appearance of the newborn ones in their proximity.

As an example, consider a maximum m that was present in the original data, which collapses together with a saddle s at time t. If s is a newborn saddle that was born at time $t' < t$ together with a maximum m', which lives longer than t, we interpret the collapse of m and s as a transition of the maximum from m to m'. We thus extend the life of m until m' dies. The life of m' could later be extended by the same mechanism, thus prolonging the life of m further, and so on.

Our goal in dealing with newborn features is therefore twofold: on one hand, to discard them, because they are not present in the original surface; on the other hand, to add their life spans in the scale-space back to the original points we want to keep, because we want to take into account every feature represented and evolving in the scale-space. To perform this task, a relationship between newborn points and original points of the same type in the scale-space must be found. As suggested in the previous example, death and birth events (which always happen in pairs, as explained in Section 2.3 and shown in Table 1) form a chain of relationships that can be exploited to recover the life span of original points. A pseudocode version of the recovery procedure is detailed in Algorithm 1. The recovered life span is computed by following the sequence of collapse and appearance events, checking and propagating the timestamps of those events along the way. The procedure stops when the critical point currently reached is alive at the end of the scale-space, or it collapses with a point that is not a newborn one.

Algorithm 1 life span recovery

```
for CriticalPoint op ∈ TrackedPoints do
  if op.is_newborn then
    continue
  end if
  if op.is_alive then
    op.lifespan = op.timestamp
    continue
  end if
  CriticalPoint current = op
  op.lifespan = 0.0
  while true do
    if op.lifespan < current.timestamp then
      op.lifespan = current.timestamp
    end if
    if current.is_alive then
      break
    end if
    CriticalPoint dm = current.death_mate
    if ¬ dm.is_newborn then
      break
    end if
    CriticalPoint bm = dm.birth_mate
    current = bm
  end while
end for
```

3.3. Final steps

All the critical points that were not present in the original, unfiltered surface are discarded, because of their spurious nature. The recovered timestamps associated with remaining points represent their life spans in the scale-space, becoming the *lifespan* values assigned to each critical point. For every maximum present, we also compute its *drop* value, using the relative drop method (Wood, 2004).

The procedures described so far can be performed as a preprocessing step and do not require user input. After that, the user selects two values for the *lifespan* and *drop* parameters to keep critical points with values equal or above the chosen thresholds. The operation can be performed with real-time responsiveness in a graphical user interface. Optionally, independent parameters can be given for the selection of peaks, pits, or saddles, to fine tune the amount of spot heights depending on their type.

4. Experimental results

Testing the results of an automated selection method requires two things: a database of manually compiled spot heights, to be used as a reference; and an assessment of how accurately results match the reference. The selection usually varies depending on one or more parameters that control the size and distribution of the resulting population. Our tests compare the proposed algorithm with known methods. Resulting spot heights are matched to the reference, using a given distance threshold as a tolerance (we used a distance $3w$, where w is the width of a cell on the DEM). The quality of the match is assessed using binary classification, which provides a well-known conceptual framework for experimental evaluation of the performance of a wide range of selection algorithms and procedures. We present test results comparing a match to a reference for the selection of peaks only. Peaks outnumber by far any other category in the reference databases that we used for testing, representing 90.3% of spot height locations associated with a toponym and 65.0% of unnamed spot height locations. Additional results that include saddles and pits are presented visually.

4.1. Binary classification

Binary classification divides a population in two different classes according to a given criterion. In our application, the input population is composed by the cells of a digital elevation model. An automated extraction method subdivides those cells into those that are spot heights and those that are not. Preexisting spot heights databases by swisstopo provide a target reference that can be used to evaluate the performance of a selection method. A perfect result would extract all the cells in the target reference and only those; with a less-than-perfect selection the elements of the input population will be classified as:

- **True Positives** (TP). A cell is extracted and it matches a reference location.
- **False Positives** (FP) A cell is extracted but it does not match a reference location.
- **False Negatives** (FN). A cell is not extracted, but a reference location is there.
- **True Negatives** (TN). A cell is not extracted, and a reference location is not there.

We adopt three basic indicators computed from these relations (Sokolova and Lapalme, 2009):

- **precision** (P), or positive predictive value, defined as $TP/(TP + FP)$.
- **recall** (R), or true positive rate, defined as $TP/(TP+FN)$.
- **specificity** (S), or true negative rate, defined as $TN/(TN + FP)$.

These indicators vary between 0 (worst result possible) and 1 (best result possible). A derived measure of the general performance of a selection method's results is the F_β score, defined as

$$F_\beta = (1 + \beta^2) \cdot \frac{P \cdot R}{(\beta^2 \cdot P) + R} \tag{5}$$

where β is a parameter that changes the relative weights of precision and recall. The F_β score varies between 0 and 1. It is a useful indicator, but it is too limited to be used as the only indicator of performance. A single number cannot fully capture all the possible behaviors of a binary classification system; a more sophisticated instrument is needed. A recommended technique is the Receiver Operating Characteristic curve (Powers, 2011).

A ROC curve plots P on the Y axis and $1 - S$ on the X axis as the parameters of the classification system vary. Unfortunately, ROC curves cannot be used in our case. The population of DEM cells is too unbalanced. True negative elements are always an overwhelming majority and, as a consequence, $1 - S$ is near zero for every possible parameter. This problem is well-known in the literature (Davis and Goadrich, 2006). When this problem arises, they suggest adoption of another widely used analysis technique, Precision-Recall curves, and show that the ROC curves and PR curves are tied by precise mathematical relations.

A PR curve is drawn by plotting precision on the Y axis and recall on the X axis as selection parameters vary. It shows the relations between the number of true positives with the number of false negatives (recall) and false positives (precision). PR space covers the unit square on the xy-plane (Figure 7). A good metric for finding the query with the best performance in PR space is identifying the point nearest to $(1, 1)$, which denotes the best possible performance.

For our evaluation, PR curves and F_β graphs are used. We chose $\beta = 0.5$, which gives twice as much importance to precision compared to recall, because we envision that most cartographers will use an automated method to achieve an

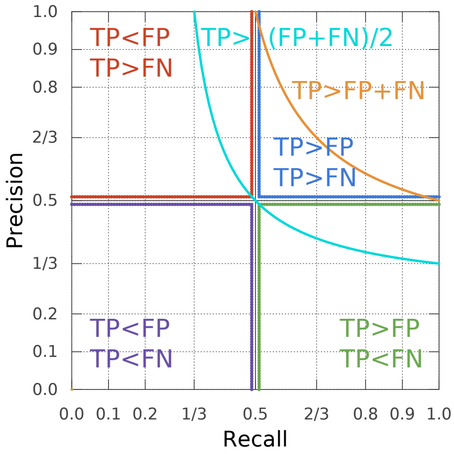


Figure 7: Graphical breakdown of Precision-Recall space. $P = 0.5$ means that $TP = FP$, and $R = 0.5$ means that $TP = FN$. A selection result with an equal number of true positive, true negative, and false negative samples is plotted at the center of PR space. The unit square is thus divided into four sectors (red, blue, violet, green). We also show the hyperbolas representing the loci of points such that the number of true positives is equal to the sum of false elements (orange, $TP = FP + FN$) and to half the sum of false elements (cyan, $TP = (FP + FN)/2$). Results reach the areas to the top-right of these functions when the number of true positive samples gradually surpasses the number of mismatches.

initial selection of spot heights that will be inspected and adjusted later. It therefore seems a good idea to favor a less cluttered selection result that has fewer false positives, even if it has more false negatives.

4.2. Test results

Two digital elevation models that include spot height locations have been used, depicting two different areas in the Swiss Alps: the *Aletsch* DEM (Figure 3) and the *Brienzersee* DEM (Figure 11). Both are sections of the swisstopo (2015) digital elevation model with a 25 m resolution. The Swiss Federal Office of Topography swisstopo provided us a database containing the spot heights placed on their 1:25,000 maps for the areas depicted in the chosen models. Reference spot heights for the Brienzersee DEM are differentiated in two classes: one class contains spot height locations associated with toponyms, which represent the most important spot heights; the other class contains unnamed spot heights locations of less importance. Details can be found in Table 2.

Data set	Size	Time	#summits	#named
Aletsch	703 × 697	24s	170	-
Brienzersee	1366 × 1134	248s	630	153

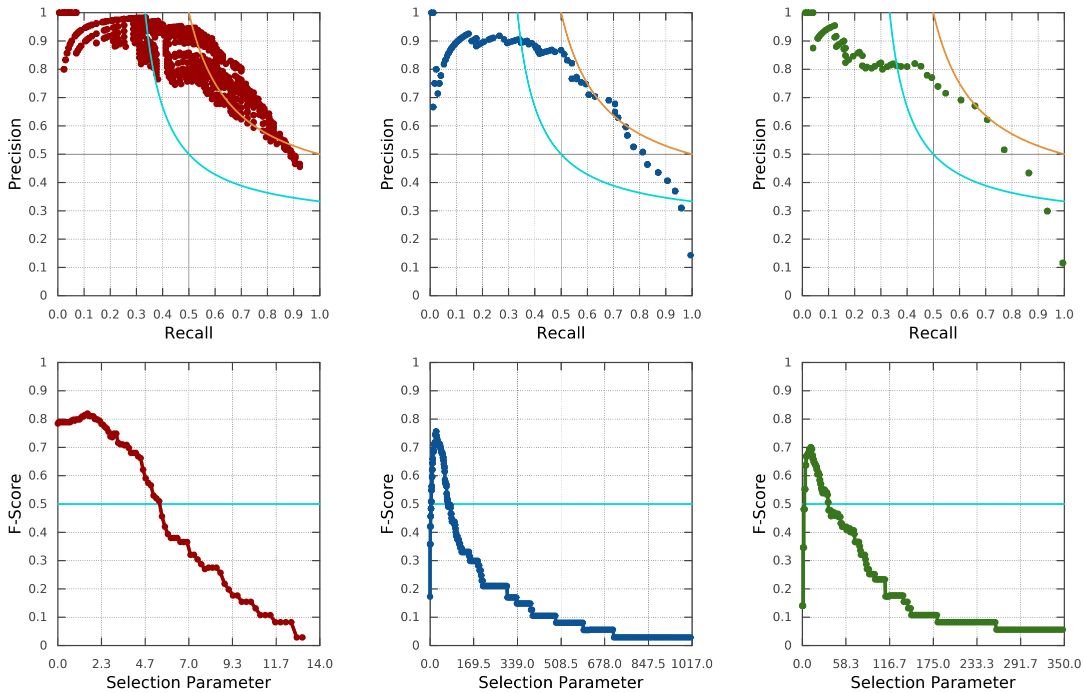
Table 2: Data sets used in the binary classification experiments, with their size (number of cells), the time elapsed for precomputation steps described in Section 3 (in seconds), the number of summits present in the 1:25,000 map, and the number of locations among summits that have a toponym.

The binary classification experimental framework could be reliably applied to the summits only. The main reason is that there are not enough data in the reference databases to carry out a good enough match and analysis for the other categories. The test area does not contain a sufficient number of depressions. Passes are present, but their number in the swisstopo database is smaller by an order of magnitude than the number of summits, and they often show a high discrepancy between the reference spot heights and the digital elevation model's surface.

For every experiment, we test our *lifespan+drop* method, described in Section 3, and two other methods as a comparison: *drop* alone, as proposed by Wood (2004), and *jaara*, the summit extraction algorithm described by Jaara and Lecordix (2011), both briefly summarized in Section 2.1. Each test consists in the execution of an automated extraction method, matched to the reference as its selection parameters vary. Evaluation is performed using PR plots and $F_{0.5}$ graphs. For the *lifespan+drop* method, which has two parameters, the PR plot is obtained by independently varying the lifespan and drop parameters for a total of 100×100 runs. The lifespan parameter covers the entire range of possible values; the drop parameter is employed as a preselection parameter, with a restricted range. The end result is a point cloud instead of a simple curve because the parameters' space is multidimensional. The $F_{0.5}$ graph is obtained by fixing the relative drop parameter to the value that gives the highest $F_{0.5}$ score in the PR plot, and by varying the lifespan parameter. The *drop* and *jaara* methods have a single parameter. For the *drop* method, it is the difference in altitude between the peak and the highest pass that must be reached in order to get to an higher peak; for the *jaara* method it is the size of the biggest square in which the peak is the highest one compared to surrounding squares (see Section 2.1). PR plots and $F_{0.5}$ graphs for *drop* and *jaara* are the result of 1000 runs with the parameter varying across the entire range of possible values. Three different experiments were carried out:

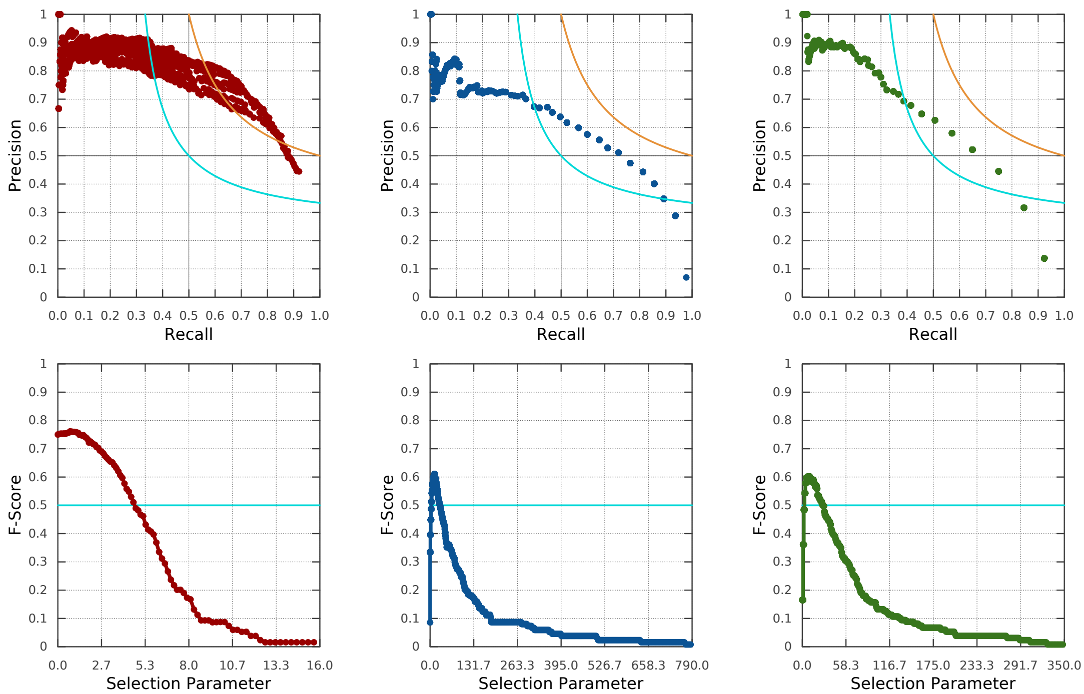
- On the Aletsch DEM, with locations classified as peaks as a reference set (Figure 8).
- On the Brienzersee DEM, using as a reference set both named summits, as provided by swisstopo, and unnamed locations classified as peaks (Figure 9).
- On the Brienzersee DEM, with named summits only as a reference set (Figure 10).

PR plots (first row of Figures 8, 9, and 10) show that *lifespan+drop* has a good advantage in the Aletsch experiment and in the Brienzersee experiment with both named and unnamed peaks, and a slight advantage in the one with named peaks only. The latter experiment is the most challenging of all three, because the reference spot heights are few and sparse. A visual rendering of the queries with the best results in PR space for the Brienzersee data set can be found in Figure 11. The $F_{0.5}$ graphs (second row of Figures 8, 9, and 10) make apparent that *lifespan+drop* provides relevant results for a longer range



life+drop drop jaara

Figure 8: Results for all peaks in the Aletsch data set. Top row: Precision-Recall plots; bottom row: $F_{0.5}$ score curves. The drop parameter in the *lifetime+drop* runs varies between 2 and 40 for the PR plot and is fixed to 13.28 for the $F_{0.5}$ curve. The best PR query has been obtained with lifetime=0.60 and drop=11.49; the best $F_{0.5}$ query with lifetime=1.58 and drop=13.28.



life+drop drop jaara

Figure 9: Results for all peaks (named and unnamed) in the Brienzersee dataset. Top row: Precision-Recall plots; bottom row: $F_{0.5}$ score curves. The drop parameter in the *lifetime+drop* runs varies between 1 and 40 for the PR plot and is fixed to 7.69 for the $F_{0.5}$ curve. The best PR query has been obtained with lifetime=0.56 and drop=5.60; the best $F_{0.5}$ query with lifetime=0.75 and drop=7.69.

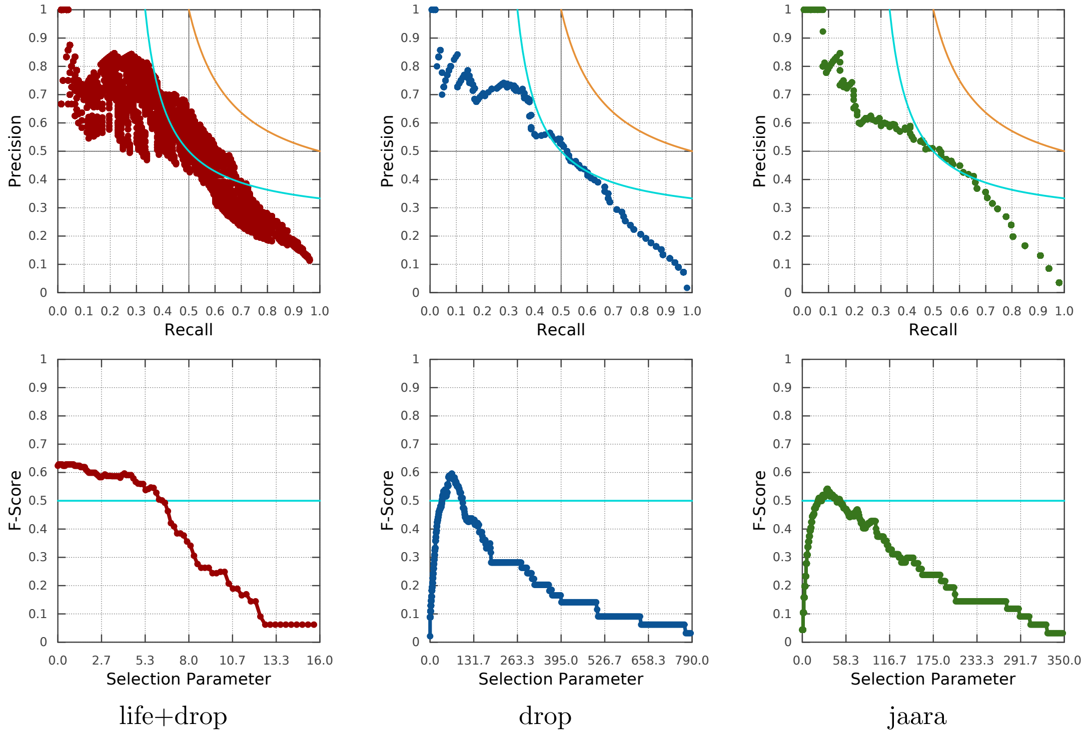


Figure 10: Results for all named peaks in the Brienzersee data set. Top row: Precision-Recall plots; bottom row: $F_{0.5}$ score curves. The drop parameter in the *lifetime+drop* runs varies between 1 and 80 for the PR plot and is fixed to 46.00 for the $F_{0.5}$ curve. The best PR query has been obtained with lifetime=0.96 and drop=40.68; the best $F_{0.5}$ query with lifetime=0.96 and drop=46.00.

of the main selection parameter when compared to the other two methods: the *drop* and *jaara* curves reach a maximum very soon and steeply fall back to low values. In the *lifespan+drop* curves, on the contrary, there is a hint of a plateau and a gentler downslope. A good visual reference is how much the $F_{0.5}$ score curve stays higher than the 0.5 value (cyan horizontal line). Note in particular how this trend is confirmed by the named peaks only experiment, which is the most challenging one.

4.3. Additional tests

We show results for two additional elevation models. The first DEM is a bathymetric data set of the ocean floor around New Zealand produced by CANZ (2008) with a 250 m resolution. We use a section of 1645 × 1466 cells (Figure 12) located north of North Island that shows an area where multibeam swath depth information has been used. The featureless flat area at the center of Figure 12 is the Raukumara plain. The second DEM is the 2000 Shuttle Radar Topography Mission (SRTM) data set with 1 arc second resolution and void areas filled by de Ferranti (2015). We use a section of 600 × 600 cells showing Monte Rosa in the Italian and Swiss Alps (Figure 2). Results are evaluated by visual inspection. For the Monte Rosa data set, we compare to a 1:50,000 map by swisstopo.

We show a selection of depressions and summits for the Raukumara plain data set in Figure 12. Most main features present in the ocean floor are properly identified and a well-balanced density of soundings is obtained, with a lesser density of soundings in flatter parts of the sea bottom. A selection of passes and summits for the eastern part of the Monte Rosa data set is shown in Figure 2. The main summits, which are situated along the impressive north-south and east-west ridges, prominently stand out and are well recognized, along with the mountain passes between them. The critical point with the longest lifespan value in the scale-space is a peak that corresponds to the Dufourspitze, the massif's highest and most relevant summit. The longest lived saddles corresponds to mountain passes that all have a toponym on the 1:50,000 Swiss map (among the three most important ones we identify the Lysjoch, which is the main trekking route across the glacier between Switzerland and Italy).

5. Conclusions

We propose an effective spot height placement method that achieves a selection dense enough in morphologically interesting regions, without selecting irrelevant points in flat areas. The method requires manipulation of two parameters at most. Rigorous experiments on peaks, which often are the most important spot heights on topographic maps, show better performance compared to other state-of-the-art methods with similar characteristics. The method is suitable to be used in a graphical user interface; after running a preprocessing step, candidate spot heights can be selected in real-time using the precomputed importance scores. We plan to develop a spot height selection GUI for

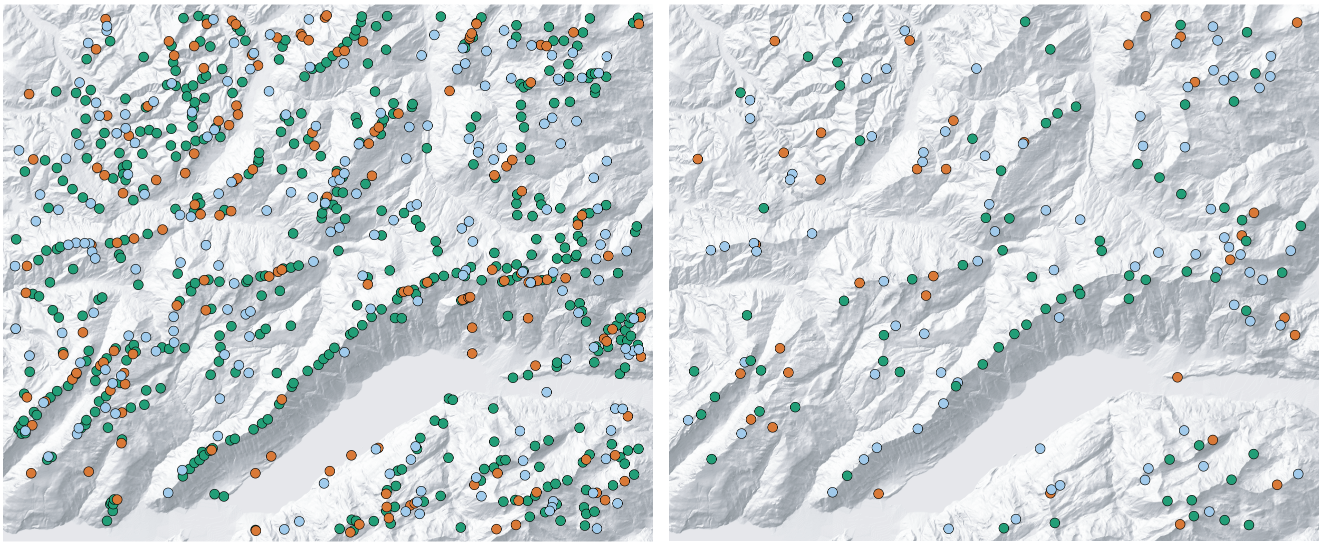


Figure 11: Best *lifetime+drop* queries in PR space for the Brienzersee data set. Green: true positive points; orange: false positive points; cyan: false negative points. Note that the sum of green and orange points represents the selection by our method, while the sum of green and cyan points represents the swiss spot heights reference. In the left image the reference is composed by named and unnamed peaks together (441 true positive points, 150 false positive points, 189 false negative points). In the right image the reference is composed by named peaks only (74 true positive points, 35 false positive points, 79 false negative points).

cartographers based on the proposed work, and to investigate whether results can be directly employed in interactive maps.

Techniques for the automatic calibration of selection parameters according to feature type, map extent, and local terrain morphology could be worth exploring. The goal is to achieve fully automated spot height placement and labeling without pre-existing reference data or human supervision.

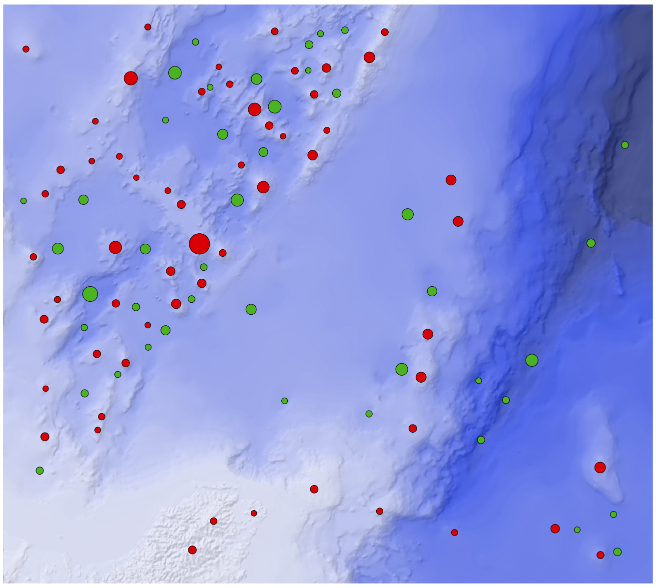


Figure 12: A selection of soundings representing summits (in red) and depressions (in green) for the Raukumara plain data set. Spot height locations have been selected using a lifespan parameter of 8. Sample points are exponentially scaled according to their lifespan value.

A promising extension of this research could be the extraction of interesting morphological features that are neither peaks nor saddles nor depressions but can be easily recognized by observers on the terrain nonetheless. Intuitively, these particular features are usually located near sudden changes of steepness, which should be characterized by high curvature. A scalar field based on curvature, such as mean or Gaussian curvature, could be computed from the digital elevation model and then directly used as input for the continuous scale-space method we described. Long-lived critical points in the curvature based scale-space that are distant enough from already selected spot heights could be good candidates for addition to large-scale topographic maps.

Acknowledgements

The authors would like to thank Jürg Gilgen, Federal Office of Topography swisstopo, and Lorenz Hurni, Institute of Cartography and Geoinformation, ETH Zürich, for their advice and help compiling reference materials for this article; Brooke E. Marston, Abby Metzger, and Gareth Baldrica-Franklin, Oregon State University, for their help with this text and figures. The authors are grateful to the anonymous reviewers for their valuable comments.

References

Baella, B., Palomar-Vázquez, J., Pardo-Pascual, J. E., Pla, M., August 2–3 2007. Spot heights generalization: deriving the relief of the Topographic Database of Catalonia at 1:25,000 from the master database. In: 11th Workshop of the ICA Commission on Generalisation and Multiple Representation, Moscow, Russia.

Baella, B., Pla, M., August 12–14 1999. Map generalization to obtain the Topographic Map of Catalonia 1:10,000. In: 3rd ICA Workshop on Progress and Developments in Automated Map Generalization, Ottawa, Canada.

CANZ, 2008. New Zealand Region Bathymetry, 1:4,000,000, 2nd Edition, URL: http://www.niwa.co.nz/our-science/oceans/bathymetry.

Comic, L., Floriani, L. D., Magillo, P., Iuricich, F., 2014. Morphological Modeling of Terrains and Volume Data. Springer Publishing Company, Incorporated.

Davis, J., Goadrich, M., 2006. The relationship between Precision-Recall and ROC curves. In: Proceedings of the 23rd International Conference on Machine Learning. ACM, pp. 233–240.

de Ferranti, J., 2015. Viewfinder Panoramas, URL: http://www.viewfinderpanoramas.org/dem3.html.

Deng, Y., Wilson, J. P., 2008. Multi-scale and multi-criteria mapping of mountain peaks as fuzzy entities. International Journal of Geographical Information Science 22 (2), 205–218.
URL http://dx.doi.org/10.1080/13658810701405623

Edelsbrunner, H., Harer, J., Zomorodian, A., 2001. Hierarchical morse complexes for piecewise linear 2-manifolds. In: Proceedings of the seventeenth annual symposium on Computational geometry. SCG '01. ACM, New York, NY, USA, pp. 70–79.
URL http://doi.acm.org/10.1145/378583.378626

Gilgen, J., 2014. Personal communication. Cartographer at the Swiss Federal Office of Topography swisstopo.

Gingold, Y. I., Zorin, D., 2007. Controlled-topology filtering. Computer-Aided Design 39 (8), 676–684.
URL http://www.sciencedirect.com/science/article/pii/S0010448507001157

Guilbert, E., Gaffuri, J., Jenny, B., 2014. Abstracting Geographic Information in a Data Rich World: Methodologies and Applications of Map Generalisation. Springer International Publishing, Ch. Terrain Generalisation, pp. 227–258, editors: Burghardt, D. and Duchêne, C. and Mackaness, W.
URL http://dx.doi.org/10.1007/978-3-319-00203-3_8

Imhof, E., 2007. Cartographic Relief Presentation. ESRI, Inc.

Jaara, K., Lecordix, F., May 2011. Extraction of Cartographic Contour Lines Using Digital Terrain Model (DTM). The Cartographic Journal 48 (2), 131–137.
URL http://dx.doi.org/10.1179/1743277411Y.0000000011

Koenderink, J. J., 1984. The structure of images. Biological Cybernetics 50 (5), 363–370.
URL http://dx.doi.org/10.1007/BF00336961

Lindeberg, T., 1994. Scale-Space Theory in Computer Vision. Kluwer Academic Publishers, Norwell, MA, USA.

Lindeberg, T., May 2011. Generalized gaussian scale-space axiomatics comprising linear scale-space, affine scale-space and spatio-temporal scale-space. J. Math. Imaging Vis. 40 (1), 36–81.
URL http://dx.doi.org/10.1007/s10851-010-0242-2

Magillo, P., De Floriani, L., Iuricich, F., 2013. Morphologically-aware Elimination of Flat Edges from a TIN. In: Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. SIGSPATIAL'13. ACM, New York, NY, USA, pp. 244–253.
URL http://doi.acm.org/10.1145/2525314.2525341

Maxwell, J. C., 1870. On hills and dales. The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science Series 4 (269), 421–427.

Milnor, J. W., 1963. Morse Theory. Vol. 51 of Annals of Mathematics Studies. Princeton University Press.

Palomar-Vázquez, J., Pardo-Pascual, J., 2008. Automated spot heights generalisation in trail maps. International Journal of Geographical Information Science 22 (1), 91–110.
URL http://dx.doi.org/10.1080/13658810701349003

Peucker, T. K., Douglas, D. H., 1975. Detection of Surface-Specific Points by Local Parallel Processing of Discrete Terrain Elevation Data. Computer Graphics and Image Processing 4 (4), 375–387.
URL http://www.sciencedirect.com/science/article/pii/0146664X75900052

Powers, D. M., 2011. Evaluation: from Precision, Recall and F-measure to ROC, informedness, markedness and correlation. Journal of Machine Learning Technologies 2 (1), 37.

Price, K., 2012. Annotated computer vision bibliography. URL: http://www.visionbib.com/bibliography/contents.html.

Rana, S., Morley, J., 2002. Surface networks.

Reininghaus, J., Kotava, N., Guenther, D., Kasten, J., Hagen, H., Hotz, I., 2011. A scale space based persistence measure for critical points in 2d scalar fields. IEEE Transactions on Visualization and Computer Graphics 17 (12), 2045–2052.

Rocca, L., Puppo, E., 2013. A virtually continuous representation of the deep structure of scale-space. In: Petrosino, A. (Ed.), Image Analysis and Processing – ICIAP 2013. Vol. 8157 of Lecture Notes in Computer Science. Springer, pp. 522–531.

Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. Information Processing and Management 45 (4), 427–437.
URL http://www.sciencedirect.com/science/article/pii/S0306457309000259

Spiess, E., 1996. Kartographie Grundzüge: Vorlesungsskript. Vol. Lecture notes in cartography, unpublished manuscript. Institute of Cartography and Geoinformation, ETH Zürich.

swisstopo, 2015. DHM25 Basis Model and Matrix Model, URL: http://www.swisstopo.admin.ch/internet/swisstopo/en/home/products/height/dhm25.html.

Takahashi, S., 2006. Algorithms for Extracting Surface Topology from Digital Elevation Models. John Wiley & Sons, Ltd, pp. 31–51.
URL http://dx.doi.org/10.1002/0470020288.ch3

Witkin, A. P., 1983. Scale-space filtering. In: Proceedings of the Eighth International Joint Conference on Artificial Intelligence - Volume 2. IJCAI'83. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 1019–1022.
URL http://dl.acm.org/citation.cfm?id=1623516.1623607

Wolf, G. W., 2006. Topographic Surfaces and Surface Networks. John Wiley & Sons, Ltd, pp. 13–29.
URL http://dx.doi.org/10.1002/0470020288.ch2

Wood, J., October 20 2004. A new method for the identification of peaks and summits in surface models. In: Proceedings of GIScience 2004—The Third International Conference on Geographic Information Science. Adelphi, MD, pp. 227–230.