C^1-continuous Terrain Reconstruction from Sparse Contours

Kai Hormann[1] Salvatore Spinello[2] Peter Schröder[1]

[1] California Institute of Technology, Department of Computer Science
[2] University College London, Department of Computer Science

Abstract

Contour lines from topographic maps are still the most common form of elevation data for the Earth's surface and in the case of historical landscapes, they often are the only available source of information. In this paper we present a new contour interpolation method that solves this bivariate problem by considering univariate curve interpolation along the approximate gradient directions of the unknown surface. For a point between two contours the height value is computed with Hermite interpolation based on the shortest distances to the contours and height and derivative information at the contours. The surfaces generated are C^1 except at terrain characteristics such as ridges and valleys which are reconstructed as sharp features. The method also faithfully reconstructs summits, pits, and saddles and is especially well-suited for sparse sets of contours. The approach allows for an efficient numerical implementation as we demonstrate with a number of examples.

1 Introduction

The surface of the Earth can be seen as a continuous bivariate scalar function. For many applications such as animation of erosion or simulation of earthquakes, this function has to be reconstructed from discrete data. The most common modern technique to obtain such terrain data is to measure the height by triangulation from an airplane or a satellite. As this usually generates data on a rectangular grid, simple and efficient algorithms such as tensor-product B-spline interpolation can be used for the reconstruction. The accuracy of modern acquisition techniques is amazing and as good as 10 by 10 m in the horizontal and 1 m in the vertical direction. The drawback of these precise measurements is the sheer size of the data that easily reaches several GB. Mesh decimation algorithms [14, 25] or subsampling can reduce the data size but special care must be taken to avoid aliasing artefacts at surface features like ridges and valleys.

A different kind of discrete data is given by the contour lines in topographic maps. In many regions of the Earth and for historical landscapes, such as volcanoes that erupted recently or glaciers that retreated during the last century, this remains the only available data source. Contour lines are either generated by the tedious task of surveying the landscape on foot, from stereographic photos, or from the measured data described above. It is hard to discuss the accuracy of contour data, but usually it is post-processed by experts who pay special attention to match the contours to surface characteristics such as ridges and valleys.

We present a new method for reconstructing terrain by interpolating a given set of contour lines with a smooth surface. The basic idea of our method is to reduce the bivariate surface reconstruction problem to univariate curve interpolation and is motivated as follows. Consider standing on the slope of a mountain at a point $p \in \mathbb{R}^2$ with a certain altitude h. Then the best strategy to descend to an elevation $h_1 < h$ and to ascend to $h_2 > h$ is to follow the surface gradient at each step. Unless this route terminates at a local extremum or a saddle point one will arrive at certain points $p_1, p_2 \in \mathbb{R}^2$ with heights h_1, h_2 and will have travelled on the *gradient path* $\Gamma \subset \mathbb{R}^2$ that connects p_1 and p_2. Estimating the height of any point $q \in \Gamma$ can therefore be seen as a univariate interpolation problem between (p_1, h_1) and (p_2, h_2).

In the case of contour interpolation, Γ is of course unknown, but we explain how to approximate it in Section 3. With this approximation at hand we then discuss the consequences of using different univariate interpolation schemes. Since not all resulting surfaces have a representation in an analytic form we show how to discretize them efficiently in Section 4.

2 Related Work

One may think of the problem of terrain reconstruction as a special case of the more general problem of 3D reconstruction from 2D cross-sections as it occurs, e.g., in the context of medical applications. There exists a significant body of literature on this topic, but most of the methods are only concerned with the reconstruction of piecewise linear surfaces and discuss how polygonal contours can be "stitched" together with triangle strips [22, 4, 16, 5] which is too restrictive for realistic terrain modelling.

The continuous version of this problem is known as *morphing* or *shape deformation* of curves. Most shape deformation techniques are either parametric or implicit. The efficient parametric methods are usually limited to transforming objects of the same topology and cannot solve the surface branching problem [28, 1, 35, 34]. Implicit methods on the other hand overcome this limitation [24, 10]. A nice approach that also handles interpolation of multiple contours was presented by Turk and O'Brien [36].

All these methods are suited for general 3D reconstruction from contours but the case of terrain reconstruction is considerably different because it requires the resulting surface to be the graph of a bivariate scalar function. In other words, it is a 2.5D reconstruction problem and the 3D methods may fail to give correct results.

A common 2.5D approach that automatically handles the surface branching problem is to set up and solve a *partial differential equation* (PDE). The Absolutely Minimizing Lipschitz Extension model [2] seems to be a promising method that reconstructs surface features very well. But as a second order PDE it cannot be used to interpolate both height and slope values at the contours and the surfaces are only C^0 along the contours. Chai et al. [8] have solved this problem by iteratively modifying the interior constraints of a second order PDE such that the surface will be C^1-continuous at the contours in the limit. But their approach is very costly to compute and cannot be used to reconstruct summits and pits. This can only be done by using forth order PDEs that allow the specification of both position and slope values at the contours. The simplest forth order PDE is the biharmonic equation. But the thin plate splines that solve this PDE tend to oscillate and are thus not suitable for realistic terrain reconstruction [18].

Similar phenomena are observed if the contour lines are treated as a set of discrete points and a *scattered data interpolation* method, e.g. radial basis functions [6, 7], bivariate splines [26], or an averaging methods such as kriging [20] and Sibson interpolation [31] is applied. These methods are widely used in geoscience to interpolate general measurement data like temperature, pressure, or wind velocity and also in computer-aided geometric design in the context of reverse engineering. But in the special case of contour interpolation, these methods do not behave very well. If the sampling of the contour lines is too dense, then the data points are not distributed regularly enough to give good results and if it is too coarse, important surface features are ignored. Dakowicz and Gold [11] therefore estimate height values for points on the medial axes of the contour lines and use them as additional data points. They compare different averaging methods but all the resulting surfaces have unwanted oscillations.

The results of our 2.5D reconstruction approach do not have such unwanted oscillations but they still feature all the properties of the high order PDE solutions. Our method handles surface branching as well as the reconstruction of summits and pits, provides a smooth C^1-continuous surface and allows for an efficient numerical implementation.

3 The Continuous Model

The problem of contour interpolation can be stated as follows. Given is a set of n contours $c_i \subset \mathbb{R}^2$ with certain heights h_i, $i = 1, \ldots, n$. Since we assume that the c_i are the iso-lines of some C^1-continuous function $f : \mathbb{R}^2 \to \mathbb{R}$ they are diffeomorphic to the circle $\mathbb{S}^1$, i.e. closed and C^1. The contours do not intersect each other[1] and are *nested*, i.e. c_i lies inside some c_j with $j < i$ for $i \geq 2$. Therefore, c_1 is the outermost contour that encloses all the others. The c_i partition the plane into $n + 1$ disjoint open sets Ω_i such that Ω_0 is the unbounded region that surrounds c_1 and the other Ω_i have c_i as an *exterior* boundary and k_i *interior* boundaries c_j with $j > i$. Note that since the c_i are the iso-lines of a continuous function, all k_i interior boundaries of Ω_i must have the same height. Figure 1 illustrates this setting. We distinguish the following cases:

[1] We do not consider the special case in which two contours intersect at a saddle point of f because this kind of degeneracy disappears for any small perturbation of the iso-value.

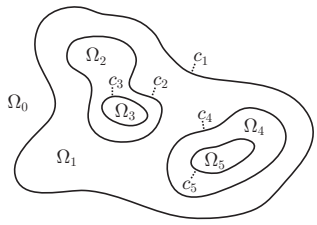


Figure 1: Five nested contours c_i with heights $h_1 < h_2 = h_4 < h_3 = h_5$ and corresponding sets Ω_i.

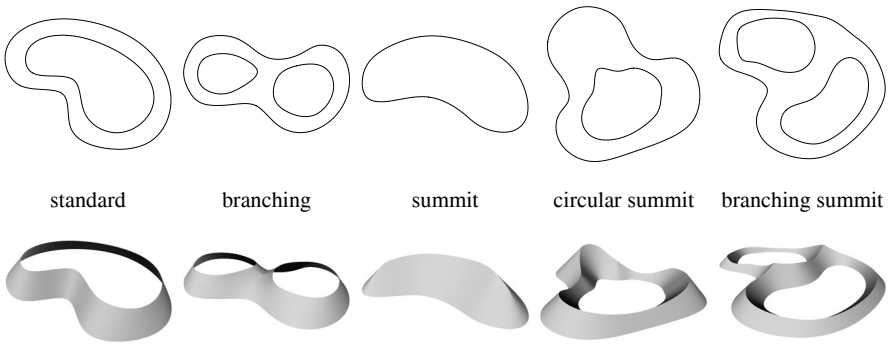


Figure 2: The different cases of terrain reconstruction.

- If $k_i = 0$, then f has a local extremum in Ω_i and we must reconstruct a *summit* or a *pit*. We distinguish between both cases by looking at the neighbouring region Ω_j which has c_i as an interior boundary. If the height of the exterior boundary c_j of Ω_j is smaller than the height of c_i, then it is a summit and otherwise it is a pit.
- If $k_i > 1$, then f has (at least) $k_i - 1$ saddle points in Ω_i and the reconstructed surface must branch from one to k_i distinct summits or pits. This case is called *surface branching*.
- For most Ω_i the exterior and interior boundaries have different heights and we call this the *standard* situation.
- If the exterior and interior boundaries have the same height, then this indicates the presence of a *circular* summit or pit. We treat this situation in the same way as normal summits and pits.

The task of reconstruction now is to find a function $h : \widehat{\Omega} \to \mathbb{R}$ with $\widehat{\Omega} = \mathbb{R}^2 \setminus \Omega_0$ that interpolates h_i at c_i and thus approximates the function f from which the contours are taken. We will now explain how to define such a function piecewise for each Ω_i such that they join with C^0- or C^1-continuity at common boundaries.

Thus, we now consider an open connected set Ω with boundaries c_1 and c_2 where each c_i is a union of disconnected components that are each diffeomorphic to $\mathbb{S}^1$. Given the associated heights $h_1 < h_2$ we then construct a function h over the closure $\bar{\Omega}$ of Ω with

$$h(p) \begin{cases} = h_1 & , \quad p \in c_1, \\ = h_2 & , \quad p \in c_2, \\ \in (h_1, h_2) & , \quad p \in \Omega, \end{cases} \tag{1}$$

so that the isolines of h for h_1 and h_2 are again c_1 and c_2. In the special case of summits, c_2 is the empty set and we replace (1) with the two conditions $h(p) = h_1$ for $p \in c_1$ and $h(p) > h_1$ for $p \in \Omega$ and similarly for pits in which case $c_1 = \emptyset$. Figure 2 shows the different cases that can occur. Note that pits are just negative summits.

3.1 Linear Interpolation

In the introduction we mentioned that we would like to find the height h for any point p in $\bar{\Omega}$ by univariate interpolation along the gradient path Γ. Albeit unknown, this path has two properties. It defines the shortest path along the surface from p to the contours c_1 and c_2 and it is perpendicular to them since the gradient $\nabla f(x)$ of a smooth function f is always orthogonal to the iso-line $\{y : f(y) = f(x)\}$.

If we now let p_1 and p_2 be those points on c_1 and c_2 that are closest to p, then the piecewise linear path $\widehat{\Gamma} = (p_1, p, p_2)$ mimics this behaviour as the line segment $[p, p_i]$ is indeed orthogonal to c_i (see Figure 3a). We therefore call $\widehat{\Gamma}$ a *first order* approximation of Γ.

The simplest kind of univariate interpolation along $\widehat{\Gamma}$ is linear interpolation (see Figure 4a),

$$h(p) = \frac{h_2 d_1(p) + d_2(p) h_1}{d_1(p) + d_2(p)}, \tag{2}$$

where

$$d_i(p) = \min_{q \in c_i} \|p - q\| \tag{3}$$

is the shortest distance from p to c_i, $i = 1, 2$. This height function obviously fulfills condition (1) because $d_i(p) = 0$ for $p \in c_i$.

To analyse the smoothness of h, we recall that the function d_i also is the solution to the *eikonal equation* $\|\nabla d_i\| = 1$ subject to the boundary condition $d_i(p) = 0$ for $p \in c_i$. It is known to be at least as

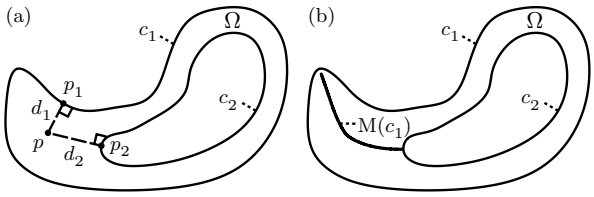


Figure 3: Closest points (a) and restricted medial axis (b).

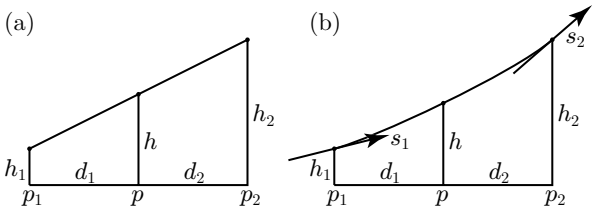


Figure 4: Linear (a) and Hermite interpolation (b) based on shortest distances.

smooth as c_i, i.e. C^1 in our case, except at the *medial axis* of c_i where it is only C^0 [27]. The medial axis $\mathrm{M}(c)$ of a planar curve c is defined as the set of those points that have more than one closest point on c (see [9] for further details). We conclude that h also is C^1-continuous except at the intersections of $\mathrm{M}(c_i)$ with Ω, $i = 1, 2$ (see Figure 3b). However, we do not consider this a restriction but rather an advantage as this nicely models ridges and valleys as sharp C^0 feature lines (see, e.g., Figure 5).

Note that this interpolation model naturally handles surface branching since we do not restrict the number of components in c_1 and c_2. In the case of summits, the distance function d_2 is not defined and we replace (2) with $h = h_1 + s\, d_1$ and likewise with $h = h_2 - s\, d_2$ for the reconstruction of pits. The constant s controls the height of the actual summit or the depth of the pit and we simply chose $s = 1$ in our examples.

3.2 Hermite Interpolation

This simple linear interpolation has also been used in [21, 32, 23] and gives good results for each individual region Ω_i. But the overall surface does not look very realistic because the contours are clearly recognizable. The reason for this is that for neighbouring regions Ω_i and Ω_j the surfaces h_i and h_j meet only with C^0-continuity at the common boundary.

One way to achieve C^1-continuity is to perform Hermite interpolation along the gradient paths instead. Assume for a moment that we are given two functions $s_i : \bar{\Omega} \to \mathbb{R}^+$ that define a *lower slope* $s_1(p)$ and an *upper slope* $s_2(p)$ for every point $p \in \bar{\Omega}$. Then we can perform Hermite interpolation as suggested by Figure 4b to find the height at p. The usual approach is to use cubic polynomials for this purpose but this can lead to height values outside the valid interval $[h_1, h_2]$. We therefore decided to use the monotonic interpolation method of Gregory and Delbourgo [19] instead. With $t_i = s_i(d_1 + d_2)/(h_2 - h_1)$ for $i = 1, 2$ and $u_1 = d_1 + t_1 d_2$ and $u_2 = d_2 + t_2 d_1$ it is given by

$$h = \frac{h_2 d_1 u_1 + u_2 d_2 h_1}{d_1 u_1 + u_2 d_2}\,. \tag{4}$$

This function clearly fulfills condition (1). By studying the gradient

$$\nabla h = \frac{(h_2 - h)\nabla(d_1 u_1) + \nabla(u_2 d_2)(h_1 - h)}{d_1 u_1 + u_2 d_2}$$

we further find that $\|\nabla h(p)\| = s_i(p)$ for $p \in c_i$ because $d_i(p) = 0$ and $\|\nabla d_i(p)\| = 1$. Note that this also holds in the case of summits or pits in which we replace (4) with $h = h_1 + s_1 d_1$ or $h = h_2 - s_2 d_2$, respectively.

This insight directly leads to a necessary and sufficient condition for the C^1-continuity of the overall surface at the given contours. Each contour c (except for the outermost one) is the common boundary of two regions, say Ω^+ and Ω^- and belongs to the sets c_1^+ and c_2^- (see Figure 6). Since both $\nabla h^+(p)$ and $\nabla h^-(p)$ are orthogonal to c for $p \in c$, we find that h^+ and h^- meet with C^1-continuity at c if the lower slope s_1^+ and the upper slope s_2^- take on the same values on c.

3.3 Slopes at the Contours

If we go back to the linear interpolation method for a moment, we see that (2) is a special case of (4) with $s_1 = s_2 = (h_2 - h_1)/(d_1 + d_2)$. It thus defines the *one-sided* slope values $\hat{s}_1 = (h_2 - h_1)/d_2$ and $\hat{s}_2 = (h_2 - h_1)/d_1$ for points on c_1 and c_2, respectively. For each contour c with neighbouring contours c_2^+ and c_1^- and associated heights $h_2^+ > h_1^+ = h = h_2^- > h_1^-$ we now take a weighted average of the one-sided slope values $\hat{s}_1^+$ and $\hat{s}_2^-$ and define

$$\bar{s} = \frac{d_2^+}{d_2^+ + d_1^-}\hat{s}_1^+ + \frac{d_1^-}{d_2^+ + d_1^-}\hat{s}_2^- = \frac{h_2^+ - h_1^-}{d_2^+ + d_1^-} \tag{5}$$

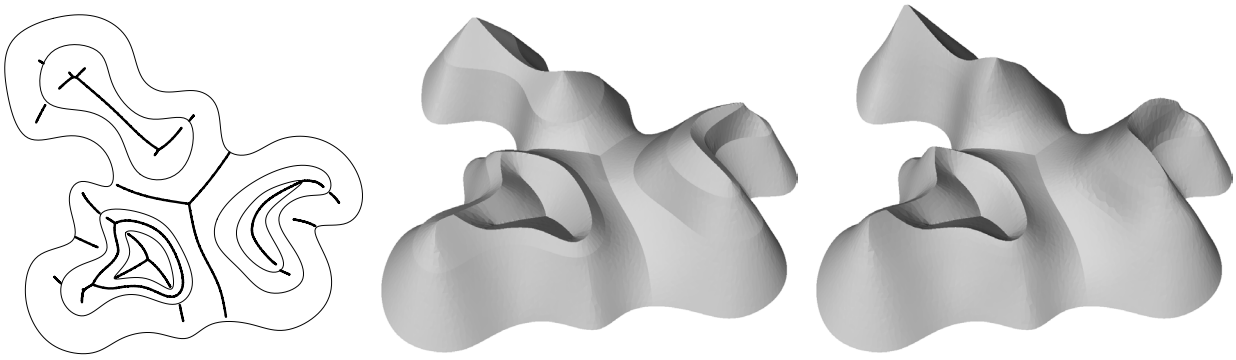


Figure 5: Reconstruction from contour lines (left) with linear (middle) and Hermite interpolation (right).

for all points on c (see Figure 6). Note that for the outermost contour and for those contours that enclose a summit or a pit, c_1^- or c_2^+ are empty and we replace (5) with $\bar{s} = \hat{s}_1^+$ or $\bar{s} = \hat{s}_2^-$, respectively.

3.4 Slopes in the Interior

After specifying the slope values on all boundaries in this way, we now have to define for each region Ω the upper and the lower slope functions s_1 and s_2 such that they interpolate $\bar{s}_1$ on c_1 and $\bar{s}_2$ on c_2, respectively. This will then guarantee the C^1-continuity of the overall reconstruction surface at the contours.

The obvious approach would be to take for each point $p \in \Omega$ the slope values at the closest points $p_i \in c_i$ (see Figure 3a) and define $s_i(p) = \bar{s}_i(p_i)$ for $i = 1, 2$. Unfortunately, these s_i are not continuous at the medial axes of c_i and the reconstructed surface has a "crack" instead of a C^0 feature line there.

We therefore decided to consider *Laplace's equation* $\Delta s_i = 0$ instead and constrained this second order PDE with the boundary conditions $s_1 = \bar{s}_1$ on c_1 and $s_1 = \hat{s}_2$ on c_2 for s_1 and likewise $s_2 = \hat{s}_1$ on c_1 and $s_2 = \bar{s}_2$ on c_2 for s_2. Figure 7 shows an example for both approaches.

As we have already mentioned, the functions h that we define with this choice of slope functions

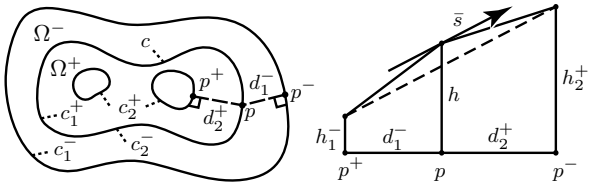


Figure 6: Slope estimation at an interior contour.

and (4) for each region Ω are C^1-continuous across common boundaries. It remains to analyse the interior smoothness. From the properties of the linear interpolation surface it follows that $\hat{s}_i$ and $\bar{s}_i$ are C^0 so that s_i as the solution of Laplace's equation with such boundary conditions is C^1 in the interior [15]. We conclude that h also is C^1-continuous except at the medial axis.

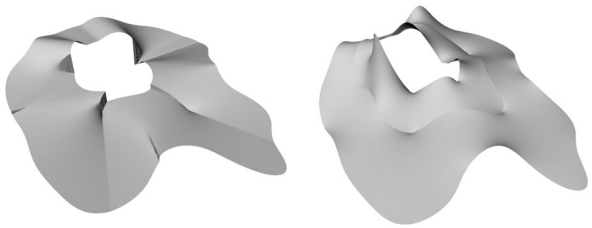


Figure 7: Functions s_1^+ and s_2^- for an intermediate contour using slopes at the closest point (left) and from Laplace's equation (right).

4 The Discrete Model

In order to display the reconstructed surface, it must be discretized in some way. We decided to use triangle meshes which are more commonly known as *triangulated irregular networks* (TIN) among geographers, but the surface could similarly be discretized over a regular grid, thus leading to a *digital elevation model* (DEM). Our implementation consists of two steps: mesh generation and surface reconstruction.

4.1 Mesh Generation

Most often the input data does not consist of smooth contour lines but rather is a set of closed polygons

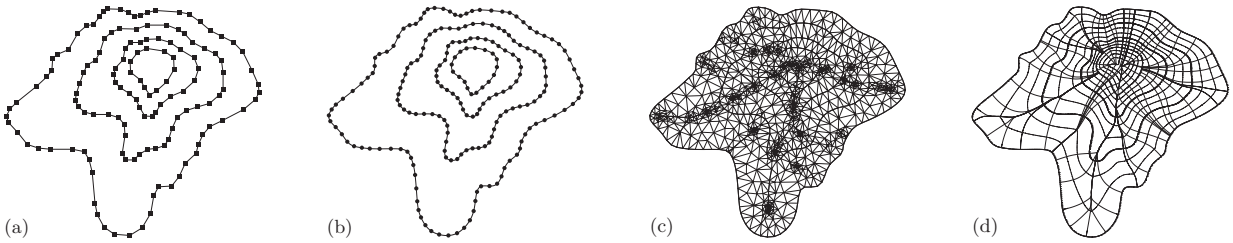


Figure 8: Input polygons (a), interpolating B-splines with regular sample points (b), triangle mesh (c), gradient paths and iso-lines of the reconstructed surface (d).

(see Figure 8a). In that case we first interpolate the vertices of each polygon with a cubic B-spline curve using the centripetal parameterization. We then discretize the contours with uniform sampling density and generate evenly spaced *contour points* (see Figure 8b). In the next step we use Shewchuk's *triangle* program [30] to create a *constrained Delaunay triangulation* (CDT) of the contour points that has an edge for each pair of consecutive contour points. We allow the program to add Steiner points in between the contours so that the triangles in the mesh are as uniform in size and shape as possible (see Figure 8c). It is exactly at these new *interior points* where we then have to compute a height value as the height of the contour points is already given by the contour heights.

In order to reduce aliasing artefacts it is desirable to have points from the medial axes of the contours as interior points. From [3] we know that if we take sufficiently dense samples from a smooth curve and consider the *Voronoi diagram* of these samples, then the collection of all the Voronoi edges whose dual Delaunay edges do not connect consecutive sample points is a good approximation of the curve's medial axis. We compute this discretized medial axis $\mathrm{M}(c)$ for each input contour c using again the triangle program and clip it against the contours c_1^- and c_2^+ in order to restrict it to the neighbouring regions Ω^- and Ω^+. The remaining edges are then taken as additional constraints in the computation of the CDT of the contour points.

4.2 Surface Reconstruction

The first step of the actual surface reconstruction is to compute for each vertex of the triangulation the shortest distances to the neighbouring contours. We accomplish this with a variant of Dijkstra's algorithm [13]. Let c be one of the boundaries of some region Ω. Then we find the distance $d(v)$ to the closest point $q(v) \in c$ for all vertices $v \in \bar{\Omega}$ as follows:

- initialize $d(v) = \infty$ for $v \in \bar{\Omega} \setminus c$ and $d(v) = 0$, $q(v) = v$ for $v \in c$
- build a priority queue Q of all $v \in \bar{\Omega}$ based on $d(v)$
- while Q not empty do
  - pop front vertex v from Q
  - for all neighbours w of v do
    - find *locally* closest point $q' \in c$ of w with $q(v)$ as initial value
    - if $d' = \|w - q'\| < d(w)$ then
      - let $d(w) = d'$, $q(w) = q'$
      - update Q

The power of this algorithm comes from the fact that it has to perform only local closest point searches with good initial values.

After computing all distance functions we determine for each region Ω the one-sided and the averaged slope values $\hat{s}_i$ and $\bar{s}_i$ for all contour points on c_i. We then use the discretization of the Laplace operator in [12] with the appropriate boundary conditions to find $s_i(v)$ for all $v \in \Omega$. This requires solving a sparse, symmetric, and positive definite linear system which can be done efficiently, e.g. with the preconditioned conjugate gradient method [17]. We finally compute the height of the reconstruction surface at all the interior vertices with (4).

5 Results

Figure 5 shows a synthetic example with all different cases that can occur. From the outermost contour the surface branches off to three different contours. Due to the shape of the northern interior contour, a double summit with a ridge in between is reconstructed. In the western region we created two neighbouring contours with the same height which creates a circular summit in between and a pit in the center. Such a shape is often found for volcanoes. The eastern summit is regular with an additional contour at greater height. Note how ridges

	synthetic	Himalayan mountain range			old St. Helens	new St. Helens	Vernagtferner
# contours	6	14			21	20	107
# contour points	865	3 950	5 840	7 892	20 944	21 366	113 725
# interior points	8 135	27 517	62 015	115 921	267 201	260 995	348 918
# triangles	17 627	61 269	133 247	244 298	572 824	561 246	925 063
medial axis (*sec*)	0.0	0.2	0.4	0.6	1.6	1.6	8.8
CDT (*sec*)	0.1	0.4	0.8	1.5	3.7	3.6	9.2
distances (*sec*)	0.2	1.1	2.5	5.6	13.0	12.3	42.6
slopes (*sec*)	1.6	4.4	15.6	37.7	87.5	86.4	121.3
overall (*sec*)	1.9	6.1	19.3	45.4	105.8	103.9	181.9

Table 1: Statistics and timings for the different examples.

and valleys are reconstructed as sharp C^0 features at the medial axes (fat lines in the left image).

Figures 9 to 13 illustrate how our method performs for real world data. In all the examples we extracted the contour lines from scanned topographic maps with the algorithm described in [33]. We carefully chose the examples in order to show that all the different reconstruction cases that we distinguish actually do occur in reality.

The model of a Himalayan mountain range in Figure 9 consists of a quarter of a million triangles and was generated from the 14 contour lines in Figure 10. This example contains multiple instances of surface branching, ridges, valleys, and summits that were reconstructed nicely to give a naturally looking landscape.

Figure 12 shows the reconstruction of Mt. St. Helens with about half a million triangles before and after the eruption in April 1980. Comparing these two models one can imagine what a devastating blast this violent display of volcanism must have been. The top two contour lines of the old Mt. St. Helens have the same height and emphasize the necessity of being able to handle circular summits. The new Mt. St. Helens proves the generality of our method that faithfully reconstructs the horseshoe top from the rather unusual crescent-shaped contour lines.

The last example in Figure 13 features the reconstruction of the Vernagtferner [2] in the Austrian Alps as it appeared in 1889. Over the last century it had to pay tribute to global warming and retreated dramatically. The topographic map from which we took the contour lines thus remains the only source of information for this historical landscape. The model consists of about one million triangles and shows that we are also able to handle *open* contours with our algorithm. We textured it with the map itself to make it look more realistic.

Table 1 finally summarizes the statistics and timings for the examples which refer to our C++ implementation of the method on an 800 Mhz Pentium III PC. The overall algorithm is approximately linear in the number of triangles with most of the time spent at solving the linear system to determine the slope values. If the number of contours and thus the proportion of contour points is large, then this last part performs slightly better because many relatively small systems need to be solved instead of a few large ones. However, the other parts of the algorithm are affected a little adversely in this case.

6 Conclusion

We presented a new, robust, and fully automatic method for terrain reconstruction from contour lines that works for all sets of nested input contours. Although we motivated our approach differently, we have seen that it is in fact PDE-based but it differs significantly from previous work. The PDE methods described in Section 2 take the reconstruction surface to be the solution of a single PDE. The surfaces that we create can be seen as *combinations* of *several* PDE solutions. By choosing the appropriate kind of combination, this leads to C^1-continuous surfaces that also reconstruct summits and pits without having to solve a high order PDE. In fact, the ingredients to our method are solutions of the first order eikonal and the second order Laplace equation both of which do not have unwanted oscillations and can furthermore be solved efficiently.

Note that the method could in principle also be used for the reconstruction of general bivariate scalar functions as long as the effect of sharp C^0 features along the medial axes is desirable.

[2] *Ferner* is an Austrian word for *glacier*.

7 Future Work

A slight drawback of our method is that the C^0 feature lines "fade out" near the contours and do not model a continuous sharp ridge or valley across the contours. The reason for this behaviour is that so far we interpolate the given data points with a C^2-continuous cubic B-spline and the medial axis $\mathrm{M}(c)$ of a contour c will therefore never touch c. In future work we will try to automatically identify feature points on the contours at which we force the interpolating curve to be C^0 only. Each of these feature points will then generate a medial axis branch that touches the contour at the feature point. Of course we also have to find a way to "guide" the medial axis branches such that they touch the neighbouring contours at feature points.

Other open questions that we will investigate on are the use of hierarchies to further speed up the computation, the approximation quality of our method, and whether it is useful for compression of terrain data. Note that we could use the exact derivative information at the contours instead of the approximation in Equation (5) in the latter application.

Acknowledgements

This work was supported in part by the Deutsche Forschungsgemeinschaft under grant HO 2457/1-1. We would like to thank Nira Dyn and David Levin for inspiring discussions about terrain reconstruction and for pointing us at the work of Gregory and Delbourgo.

References

[1] M. Alexa, D. Cohen-Or, and D. Levin. As-rigid-as-possible shape interpolation. In *Siggraph 2000 Conference Proceedings*, pages 157–164, 2000.

[2] A. Almansa, F. Cao, Y. Gousseau, and B. Rougé. Interpolation of digital elevation models using AMLE and related methods. *IEEE Transactions on Geoscience and Remote Sensing*, 40:314–325, 2002.

[3] N. Amenta, M. Bern, and D. Eppstein. The crust and the β-skeleton: Combinatorial curve reconstruction. *Graphical Models and Image Processing*, 60:125–135, 1998.

[4] C. L. Bajaj, E. J. Coyle, and K. N. Lin. Arbitrary topology shape reconstruction from planar cross sections. *Graphical Models and Image Processing*, 58:524–543, 1996.

[5] G. Barequet, M. T. Goodrich, A. Levi-Steiner, and D. Steiner. Straight-skeleton based contour interpolation. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 119–127, 2003.

[6] R. K. Beatson and W. A. Light. Fast evaluation of radial basis functions: Methods for two-dimensional polyharmonic splines. *IMA Journal of Numerical Analysis*, 17:343–372, 1997.

[7] M. Buhmann. Radial basis functions: the state-of-the-art and new results. *Acta Numerica*, 9:1–37, 2000.

[8] J. Chai, T. Miyoshi, and E. Nakamae. Contour interpolation and surface reconstruction of smooth terrain models. In *Proceedings of the Conference on Visualization '98*, pages 27–33, 1998.

[9] H. I. Choi, S. W. Choi, and H. P. Moon. New algorithm for medial axis transform of plane domain. *Graphical Models and Image Processing*, 59:463–483, 1997.

[10] D. Cohen-Or, D. Levin, and A. Solomovici. Three dimensional distance field metamorphosis. *ACM Transactions on Graphics*, 17:116–141, 1998.

[11] M. Dakowicz and C. M. Gold. Extracting meaningful slopes from terrain contours. In *International Conference on Computational Science*, pages 144–153, 2002.

[12] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Siggraph '99 Conference Proceedings*, pages 317–324, 1999.

[13] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[14] M. Garland and E. Shaffer. A multiphase approach to efficient surface simplification. In *Proceedings of the Conference on Visualization '02*, pages 117–124, 2002.

[15] D. Gilbarg and N. S. Trudinger. *Elliptical Partial Differential Equations of Second Order*. Springer, 2001.

[16] C. Gitlin, J. O'Rourke, and V. Subramannian. On reconstructing polyhedra from parallel slices. *International Journal of Computational Geometry and Applications*, 6:103–122, 1996.

[17] G. H. Golub and C. F. van Loan. *Matrix Computations*. The John Hopkins University Press, 1989.

[18] M. Gousie and R. Franklin. Converting elevation contours to a grid. In *Proceedings of the 8th International Symposium on Spatial Data Handling*, pages 647–656, 1998.

[19] J. A. Gregory and R. Delbourgo. Piecewise rational quadratic interpolation to monotonic data. *IMA Journal of Numerical Analysis*, 2:123–130, 1982.

[20] A. G. Journel and C. H. Huijbregts. *Mining Geostatistics*. Academic Press, 1981.

[21] K. Kaneda, F. Kato, E. Nakamae, T. Nishita, and T. Noguchi. Three dimensional terrain modeling and display for environment assessment. *Computer Graphics*, 23:207–214, 1989.

[22] E. Keppel. Approximating complex surfaces by triangulation of contour lines. *IBM Journal of Research and Development*, 19:2–11, 1975.

[23] J. S. Lee and S. J. Chung. Reconstruction of 3d terrain data from contour map. In *MVA '94 IAPR Workshop on Machine Vision Applications*, pages 281–284, 1994.

[24] D. Levin. Multidimensional reconstruction by set-valued approximation. *IMA Journal of Numerical Analysis*, 6:173–184, 1986.

[25] P. Lindstrom. Out-of-core simplification of large polygonal models. In *Siggraph 2000 Conference Proceedings*, pages 259–262, 2000.

[26] G. Nürnberger and F. Zeilfelder. Developments in bivariate spline interpolation. *Journal of Computational and Applied Mathematics*, 121:125–152, 2000.

[27] H. Pottmann and J. Wallner. *Computational Line Geometry*. Springer, 2001.

[28] T. Samoilov and G. Elber. Self-intersection elimination in metamorphosis of two-dimensional curves. *The Visual Computer*, 14:415–428, 1998.

[29] B. Schneider. Geomorphologically sound reconstruction of digital terrain surfaces from contours. In *Proceedings of the 8th International Symposium on Spatial Data Handling*, pages 657–667, 1998.

[30] J. R. Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry: Theory and Applications*, 22:21–74, 2002.

[31] R. Sibson. A vector identity for the Dirichlet tesselation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 87:151–155, 1980.

[32] P. Soille. Spatial distribution from contour lines: An efficient methodology based on distance transformations. *Journal of Visual Communication and Image Representation*, 2:138–150, 1991.

[33] S. Spinello and G. Greiner. Automatic contour line recognition from scanned topographic maps. Technical report, University of Erlangen, 2001.

[34] T. Surazhsky and G. Elber. Metamorphosis of planar parametric curves via curvature interpolation. *International Journal of Shape Modeling*, 8:201–216, 2002.

[35] T. Surazhsky, V. Surazhsky, G. Barequet, and A. Tal. Blending polygonal shapes with different topologies. *Computers & Graphics*, 25:29–39, 2001.

[36] G. Turk and J. F. O'Brien. Shape transformation using variational implicit functions. In *Siggraph '99 Conference Proceedings*, pages 335–342, 1999.

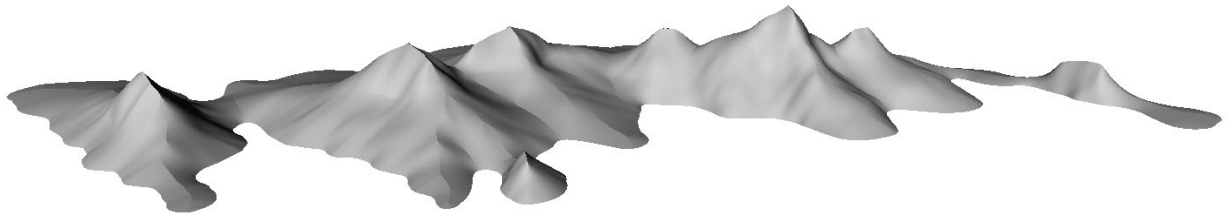


Figure 9: Terrain reconstruction from the contours in Figure 10.

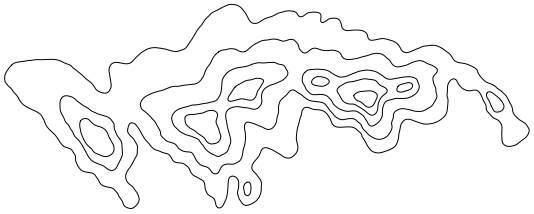
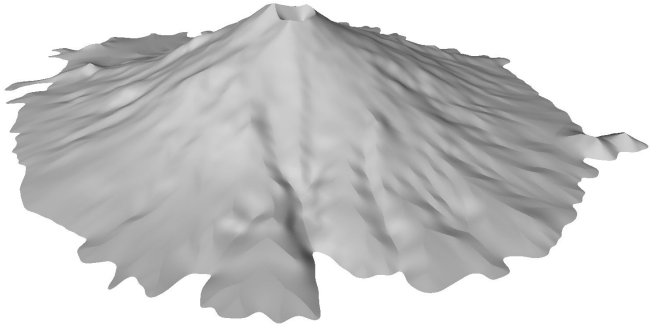


Figure 10: Contour lines of a mountain range.



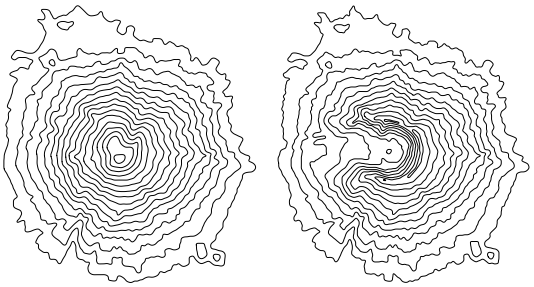


Figure 11: Contour lines of Mt. St. Helens before (left) and after (right) the eruption.

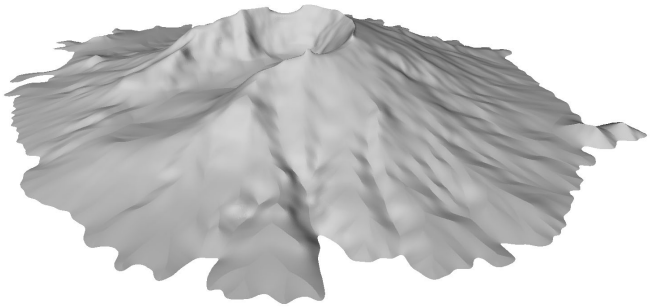
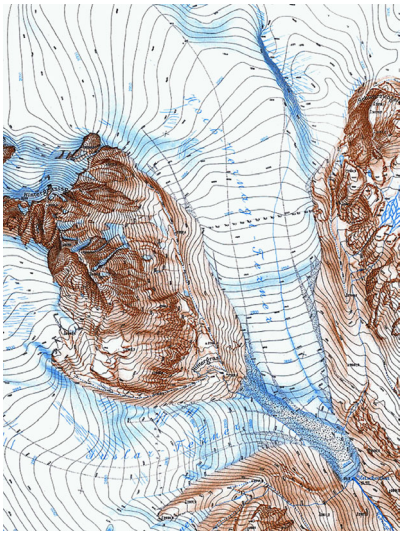


Figure 12: Terrain reconstruction from the contours in Figure 11.



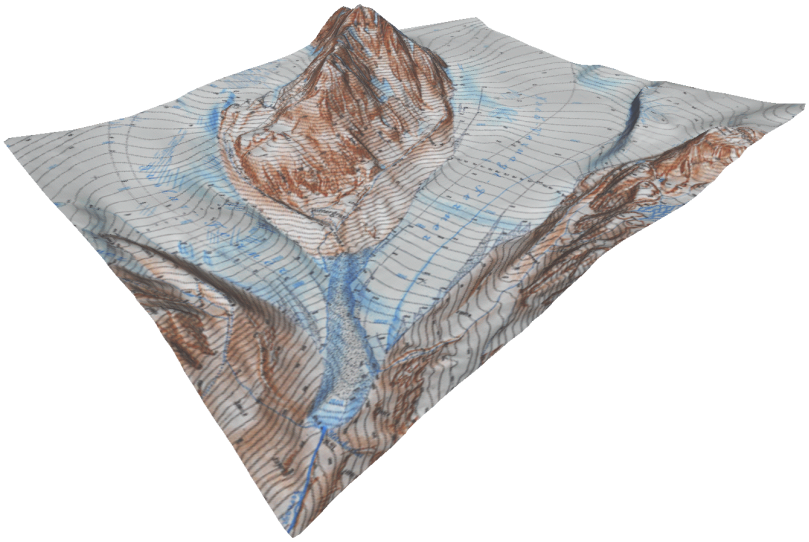


Figure 13: Terrain reconstruction of the Vernagtferner in 1889 (right) from contour lines that were extracted from a scanned map (left).