

Creating visually consistent DEMs over water bodies using laser scanning data

Ville Mäkinen
Finnish Geospatial
Research Institute FGI,
National Land Survey of
Finland
Masala, Finland
ville.p.makinen@nls.fi

Juha Oksanen
Finnish Geospatial
Research Institute FGI,
National Land Survey of
Finland
Masala, Finland
juha.oksanen@nls.fi

Tapani Sarjakoski
Finnish Geospatial
Research Institute FGI,
National Land Survey of
Finland
Masala, Finland
tapani.sarjakoski@nls.fi

Abstract

The digital elevation model (DEM) is an invaluable product in numerous geospatial applications from orthorectification of aerial photographs to hydrological modelling and advanced 3D visualisation. With the current aerial laser scanning methods, superior quality digital elevation models can be produced over land areas, but surfaces over water bodies are visually problematic, especially for streams in 3D. We present a method to generate smooth, monotonically decreasing elevation surfaces over water bodies in DEMs. The method requires the point cloud data and the polygons delineating the water bodies as input data. We show how DEM visualisations improve by applying the presented method.

Keywords: Algorithms, modelling, visualization

1 Introduction

Advances in Light Detection and Ranging (LiDAR) technology and in Airborne Laser Scanning (ALS) (Wehr & Lohr, 1999) make it possible to produce more accurate and denser point clouds. ALS point clouds are already used in numerous applications, for example in vegetation mapping (Mason, 2003), forest inventory (White, 2013), and building detection (Dorninger & Pfeifer, 2008). One of the most important products that can be derived from ALS data is the digital elevation model (DEM), which is a digital representation of the ground surface. DEMs are often generated from the ALS point cloud by first generating a triangulated irregular network (TIN) and then either aggregating or interpolating it into a regular grid, depending on the resolution of the grid and the density of the point cloud.

Water bodies (lakes, streams) are problematic for current ALS technology. The laser pulse returns from the surface of the water are often faint or missing completely, and the measured point clouds usually have lower point density and higher noise over water bodies (Höfle et al, 2009). This propagates into the DEMs as unnaturally rough surfaces and visible triangles over the water bodies, which limits the usability of the DEMs in visualisations.

The common way to overcome the problem is to mask the water bodies after the DEM is visualised. For 2D visualisations, this approach is adequate for many purposes. Exceptions are areas where the slope of the water surface is steep (for example, in rapids), because they will not be visible in representations that use the slope of the surface as a parameter. These include, for example, analytical hillshading (Imhof, 1982). However, different 3D visualisation techniques, especially in web browsers, are improving all the time and are becoming a mainstream visualisation method. The 3D visualisation of lakes may be improved by setting

their elevation to a suitable constant value, but gradually lowering surfaces are required for streams. Currently there is no well-established method to generate such surfaces from the point cloud data.

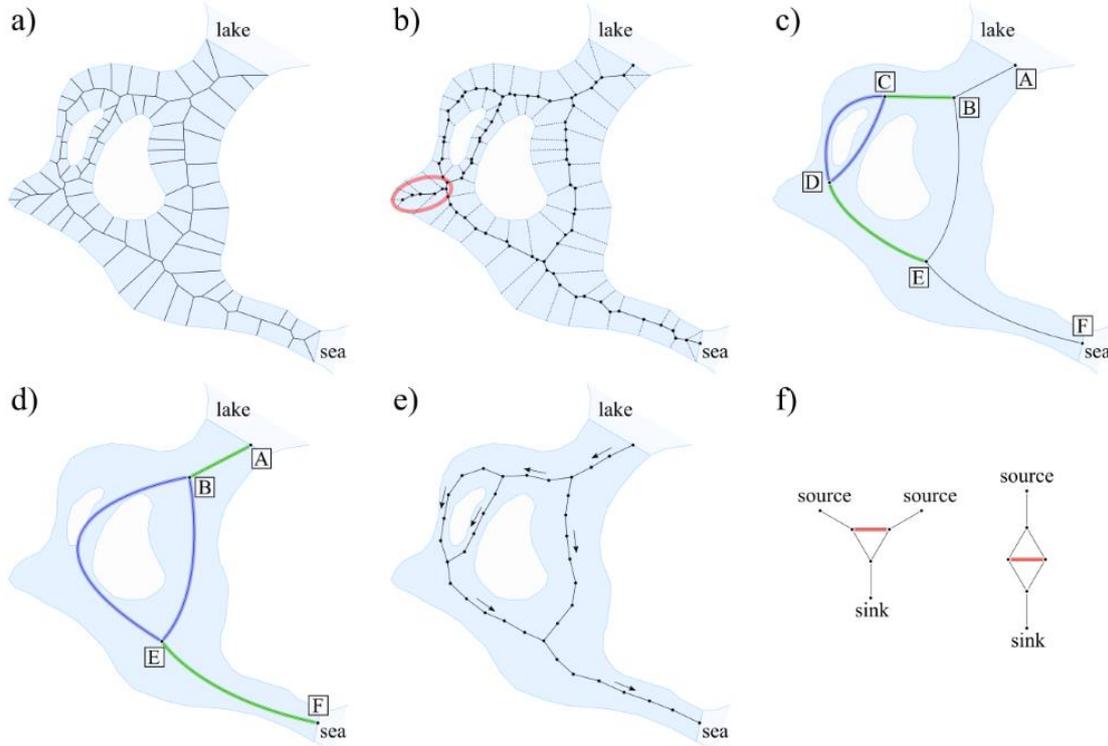
In this article we first present a method to approximate the flow directions for arbitrary stream polygons. We then propose a method to generate visually pleasing elevation surfaces for the streams and show how DEM visualisations may be improved by applying the presented method.

2 Materials and Methods

The method we have developed requires ALS point cloud data of the area and vector data (polygons) representing the lakes and the streams of the study area. The use of supporting polygon data is justified, because if such data is not available, methods exist to extract the water areas from the point cloud data (Höfle et al, 2009). We use point cloud data produced by ALS measurement of the National Land Survey of Finland (NLS ALS, 2019). The point cloud is manually classified and contains at least 0.5 points/m². The polygons covering the water areas are taken from the Topographic Database (NLS Topo, 2019).

The hydrological network formed by lakes and streams connected to each other may extend over continents. Processing such a huge network at the same time is challenging, partly because of the computational demands. We therefore split the problem into smaller parts and process them separately. We start the process with the lakes, and then move on to processing the streams. The focus of this work is on the streams. We therefore process the lakes by extracting a suitable elevation for each lake from ALS data and rasterise the lake polygons by filling them with the extracted elevations.

Figure 1: The process for determining flow directions for a stream polygon. (a) A fictitious stream polygon and a schematic raw skeleton. (b) The skeleton with unnecessary parts removed and the final skeleton connected to the lake and sea polygons. (c) A graph formed from the cleaned skeleton. The two edges between C and D are treated as one, which further allows us to join the segments B-C, C-D, and D-E as a single edge. (d) A simplified graph to which the directions may be assigned unambiguously. (e) The flow directions determined for each graph edge. (f) Two simple examples of graphs for which the direction of the highlighted edge



For the streams, we aim to create elevation surfaces whose elevation decreases monotonically along their flow direction but stays constant along their cross-sections. Adjusting the stream elevations in this manner also improves the use potential of the DEMs in aspects other than merely visualisation.

The main steps of our approach to create smoothed elevation surfaces over the streams are to a) create a simplified line representation, or graph, for the stream polygon, b) assign the flow directions for the edges of the graph, and c) use the directed graph to produce the final smooth surface for the polygon.

2.1 Extraction of stream skeletons

We start the process by generating a skeleton for the stream polygon. The aim is to create the skeleton as simply as possible, but also contain enough features to describe the overall flow inside the polygon in an adequate level of detail. The stream may have islands, which are represented as holes in the polygon. However, small islands do not affect the flow directions considerably. Therefore, all the islands whose area is below a given threshold are removed from the polygon before the skeleton is created.

To generate a skeleton for the stream polygon, we use the 2D straight skeleton method as it is implemented in the CGAL package (The CGAL Project). The resulting raw skeleton often contains many unnecessary segments that are not needed in this use case (Figure 1a). First, we exclude all the segments that are connected to the boundary of the polygon, which already simplifies the skeleton considerably (Figure 1b).

Many of the small coves and bays in the stream polygon give rise to segments one of whose ends is not connected to the others and which are not close to lake or sea polygons (Figure 1b, the circled segment). Such “dangling” segments are usually not important for determining the principal flow directions. However, the stream polygons may have long, narrow parts that are not connected to any lake or sea, but whose shape suggests they are an important part of the polygon. Blindly removing all the dangling segments would also erase these features. We decided to remove only the dangling segments for which a vector connecting the two end vertices is contained in the stream polygon (circled in Figure 1b). After the exclusion is done, we connect the skeleton vertices to the connected lake and sea polygons.

This part of the process is closely related to the generalisation where similar approaches have been used to create line representations of polygons for smaller scales (Haunert & Sester, 2008). Generalisation may be performed with different criteria that depend on the specific use case. Here the main goal is to capture the principal topology of the object and its neighbours.

2.2 Determination of the graph edge directions

We need to assign a flow direction for each segment in the cleaned skeleton. We begin the process by treating the skeleton as a graph. Each vertex connected either to one or to more than two segments is taken as a node of the graph, and the linear segment sequences between them as the edges of the graph (Figure 1c). The nodes connected only to a single edge are called leaf nodes.

We define two special classes of leaf nodes. A node is a sink if only edges are directed into it. Similarly, a node is a source if all the connected edges originate from it. We assign the special nodes and edge directions according to the following rules:

1. If the graph is connected to the sea, all the nodes touching the sea polygons are classified as sinks, and the other leaf nodes are classified as sources.
2. If the graph is only connected to a single lake, the connecting nodes are classified as sinks, and the other leaves as sources.
3. If there are many connecting lakes, the nodes connected to the lowest lake are set as sinks, and the other leaf nodes as sources.
4. For each non-leaf node, edges must be directed both inwards to and outwards from the node.
5. The graph must not contain loops. In other words, it is not permissible to start from any node and end up at the same node by following the directed edges.

These simple rules are often enough to assign the directions for edges unambiguously.

An illustrative example of the direction assignment is shown in Figure 1(c-e). Node A is a source and node F is a sink (rule 1). Because loops are not allowed (Figure 1c), the two edges connecting nodes B and C must be directed in the same way, and we can collapse them into a single edge. Then the sub-graph (B-C-D-E) may only have two valid directional configurations, and it may also be merged into a single edge (Figure 1d). Finally, the nodes B and E are now connected by two edges that may again be collapsed into one. This leaves us with a simple graph with the only leaf nodes being the sink and the source, and the directions for the edges, and subsequently for skeleton segments, may be assigned unambiguously (Figure 1e).

There are cases where the simple rules presented above are insufficient to determine the edge directions unambiguously for the emerging graphs. The two simplest examples of such graphs are shown in Figure 1f, where a valid graph can be produced regardless of the direction of the highlighted edge. This would be the case in the previous example if the circled skeleton segment in Figure 1b could not be removed. In such cases, we inspect the elevation values and estimate the average slopes of the segments to determine the flow direction. In low-relief areas, this may be difficult because the elevation differences are often minuscule and may be lost in

the noise in the data. On the other hand, in such cases, choosing the wrong flow direction does not necessarily result in a dramatic error. If these methods need to be used to determine the flow directions for some segments, the final graph must be validated against the above rules (especially rules 4 and 5).

2.3 Rasterisation by skeleton expansion

Previously, we specified that the elevation surfaces for the streams should stay constant along the cross-sections of the streams and decrease monotonically along the flow directions. We have examined a way to achieve this by gradually expanding the directed skeletons.

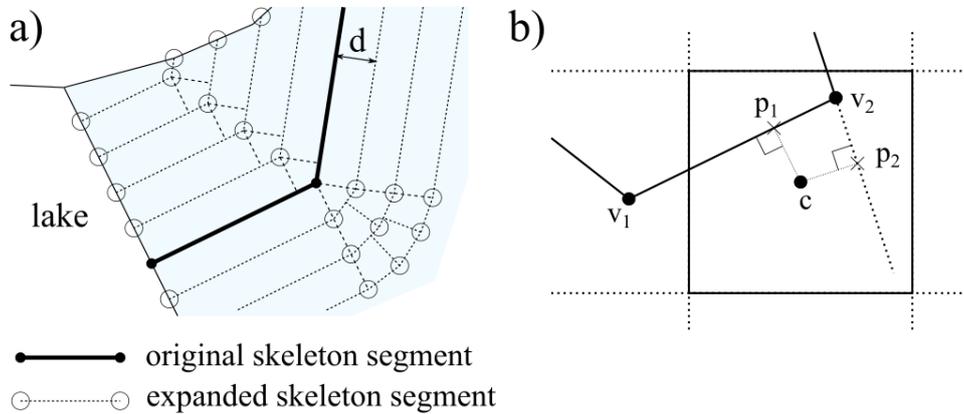
We start the process by approximating the elevations of the skeleton vertices. For the vertices connected to a lake, we choose the elevation that we previously determined for that lake. For other vertices, we approximate the elevation using ALS data. Once the elevations are determined, we travel all the paths from the sources to the sinks, and lower the elevations of the vertices so that none of the segments flow uphill. The result is a monotonically decreasing elevation profile along the flow directions.

Once the elevations are adjusted, we expand the skeleton with parallel lines at regular intervals (Figure 2a). Distance d between the lines must be slightly smaller than the resolution of the grid to ensure that the lines intersect most of the cells. The expansion is executed iteratively, and at each iteration the expanded lines are clipped with the stream polygon. This ensures the expansion does not jump over land areas.

The elevations for the vertices of these new lines are defined as the average of the elevations of the closest points on the lines of the previous layer. The elevations on the closest points of the line segments are determined by a linear interpolation.

With the skeleton and the new parallel lines, we can begin rasterisation. We assign an elevation value for each cell that the segments intersect. The values for the cells are obtained as follows: we project the centre point of the cell onto the intersecting segment, and obtain the value of that point by linear interpolation between the elevations of the vertices of the segment (Figure 2b). If the projected point does not touch the segment, the elevation of the closest vertex is used. If multiple segments intersect the cell, the minimum of the values is chosen.

Figure 2: An example of the skeleton expansion algorithm. (a) The original skeleton is expanded perpendicularly to the segments. The elevation values for the new vertices are obtained by finding the closest point in the previous skeleton and calculating the value of that point by linear interpolation between the vertices. If spacing d between lines is small enough, rasterising the lines assigns a value for most of the cells under the stream polygon. (b) The elevation for the cell is determined by projecting its centre point c onto the skeleton segments (p_1 and p_2) and choosing the smallest of the projected elevations. The value for p_1 is obtained by linear interpolation between v_1 and v_2 ; for p_2 the value of v_2 is chosen. Note that the segments are usually much longer than the cell size of the raster.

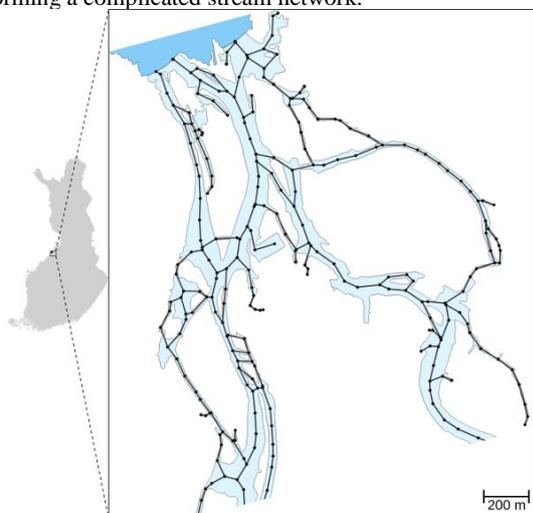


3 Results and Discussion

3.1 The directions of the skeleton segments

The assignment of the directions for the skeleton segments is based on a few simple rules. For simple stream polygons, the method works well. For braided streams and river deltas, where the streams may bifurcate and unite erratically (Figure 3), the result is a complex skeleton that may have several segments with ambiguous directions. In low-relief areas, the elevations of the skeleton vertices may also be almost identical, which renders them unreliable for determining the directions. It may therefore be useful to manually inspect the directed skeletons the method generates.

Figure 3: Part of a river delta of Pyhäjoki, Finland, forming a complicated stream network.



3.2 Rasterisation of streams

The skeleton expansion method presented here works well for linear streams. Both 2D and 3D visualisations are improved by using the smooth surface mask for the stream (Figures 4 and 5). Areas where three skeleton segments join may cause problems if the slopes of the segments are considerably different. In such cases, sharp edges may be generated in between the segments. One way to remedy the situation would be to set the elevation of the vertices near the nodes to identical values. Sometimes this may lead into large flat areas, even if the skeleton segments originally have non-zero slopes.

In our approach, we generate the stream cross-sections implicitly by expanding the skeleton iteratively. Another approach would be to attempt to generate the cross-sections explicitly, adjust their values using the directed skeleton, and then use the cross-sections to generate the raster for the stream. This would be a straightforward process for simple linear stream sections, but for more complex sections generating the cross-sections is complicated.

3.3 Applicability

The point cloud data used in this work is manually classified, which makes working with it easier. However, in principle any point cloud data can be used. The density of the point cloud determines the largest scale where the process can perform reasonably.

So far, we have focused on a single resolution when generating the DEM and the stream mask. Therefore, we have used fixed values for parameters such as the threshold to select small islands to be removed from the streams. For smaller scales larger island should be removed, and in general the values for the parameters should depend on the used scale.

The presented method should work from flat to steep areas. On very flat areas the method reduces to masking the streams with a constant elevation. On areas where the slope of a stream varies considerably, the skeleton vertex density may have to be increased in order the skeleton to model the surface realistically.

Figure 4: (a) A DEM of 1-metre resolution, generated directly from the laser scanning data. (b) The DEM is enhanced with the water mask. In both images, the DEM is coloured with greyscale values (light=high, dark=low) and with a hillshade.

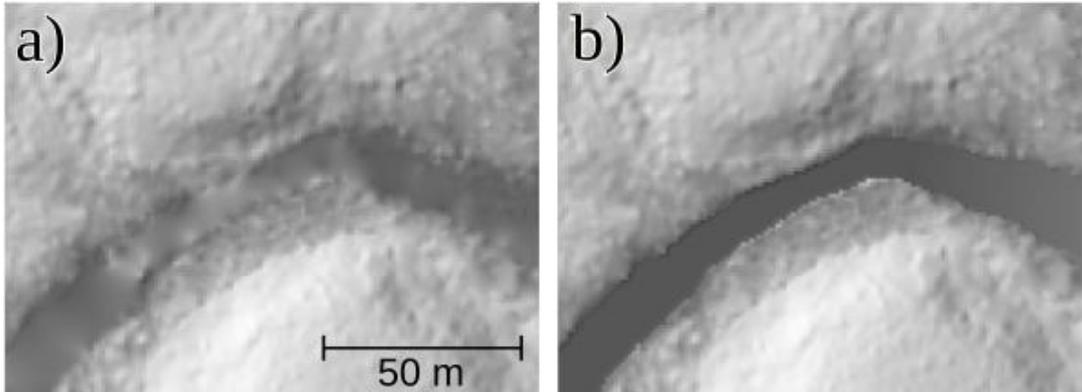
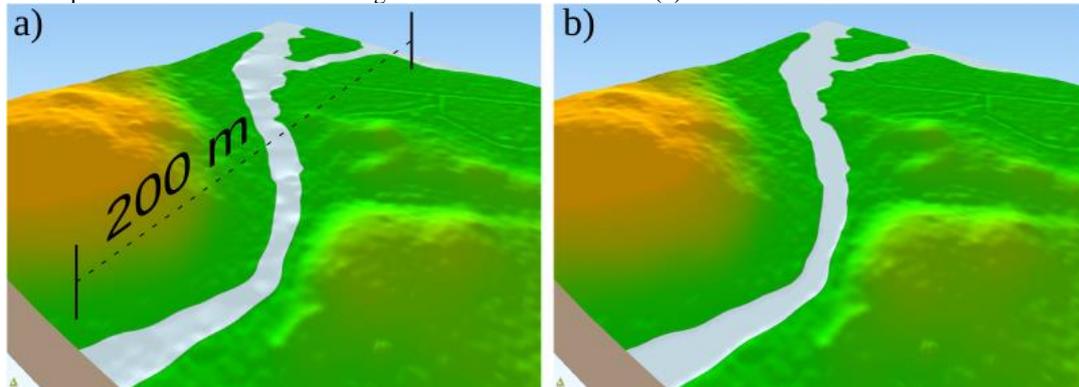


Figure 5: 3D visualisation of the DEM in 1-metre resolution. (a) The DEM generated directly from the TIN by linear interpolation. There are visible triangular artefacts in the stream. (b) The DEM where the stream is smoothed.



There are cases, such as rapids, where the surface of a stream is not smooth. Currently the method smoothens them out as well, but it could be possible to include additional information to switch the method off on those locations.

4 Conclusions

We have developed a method to create smooth elevation surfaces for streams, which are often unnaturally rough in the DEMs generated from ALS data. The method consists of determining approximate flow directions for the stream polygons and using the flow direction data to generate a smooth, monotonically decreasing elevation surface covering the streams. The main application is to improve 3D visualisations especially. These are gaining popularity due to the improvement of web technologies. We have demonstrated that the presented method improves the DEMs over the streams, and that this translates into better visualisations in both 2D and 3D.

References

- Dorninger, P. and Pfeifer, N. (2008) A Comprehensive Automated 3D Approach for Building Extraction, Reconstruction, and Regularization from Airborne Laser Scanning Point Clouds. *Sensors*, 8, 7323-7343.
- Hauert, J.-H. and Sester, M. (2008) Area Collapse and Road Centerlines based on Straight Skeletons. *GeoInformatica*, 12(2), 169-191.
- Höfle, B., Vetter, M., Pfeifer, N., Mandlbürger, G. and Stötter, J. (2009) Water surface mapping from airborne laser scanning using signal intensity and elevation data. *Earth Surface Processes and Landforms*, 34, 1635-1649.
- Imhof, E. *Cartographic Relief Presentation*. de Gruyter, Berlin, 1982.
- Mason, D. C., Cobby, D. M., Horritt, M. S. and Bates, P. D. (2003) Floodplain friction parametrization in two-dimensional river flood models using vegetation heights derived from airborne scanning laser altimetry. *Hydrological Processes*, 17, 1711-1732.
- NLS ALS (2019) Laser scanning data, National Land Survey of Finland. Available at <https://www.maanmittauslaitos.fi/en/maps-and-spatial-data/expert-users/product-descriptions/laser-scanning-data>
- NLS Topo (2019) Topographic Database, National Land Survey of Finland. Available at <https://www.maanmittauslaitos.fi/en/maps-and-spatial-data/expert-users/product-descriptions/topographic-database>

The CGAL Project. *CGAL User and Reference Manual*.
CGAL Editorial Board, 4.13 edition, 2018.

Wehr, A. **and** Lohr, U. (1999) Airborne laser scanning – an
introduction and overview. *ISPRS Journal of
Photogrammetry & Remote Sensing*, 54, 68-82.

White, J. C., Wulder, M. A., Vastaranta, M., Coops, N., Pitt,
D. **and** Woods, M. (2013) The Utility of Image-Based
Point Clouds for Forest Inventory: A Comparison with
Airborne Laser Scanning, *Forests*, 4, 518-536.