Automatic identification of watercourses in flat and engineered landscapes by computing the skeleton of a LiDAR point cloud

Broersen, Tom; Peters, Ravi; Ledoux, Hugo

DOI
10.1016/j.cageo.2017.06.003

Publication date
2017

Document Version
Accepted author manuscript

Published in
Computers & Geosciences: an international journal

Citation (APA)
Broersen, T., Peters, R., & Ledoux, H. (2017). Automatic identification of watercourses in flat and engineered landscapes by computing the skeleton of a LiDAR point cloud. *Computers & Geosciences: an international journal*, *106*, 171-180. https://doi.org/10.1016/j.cageo.2017.06.003

Automatic identification of watercourses in flat and engineered landscapes by computing the skeleton of a LiDAR point cloud

Tom Broersen[a], Ravi Peters[a,*], Hugo Ledoux[a]

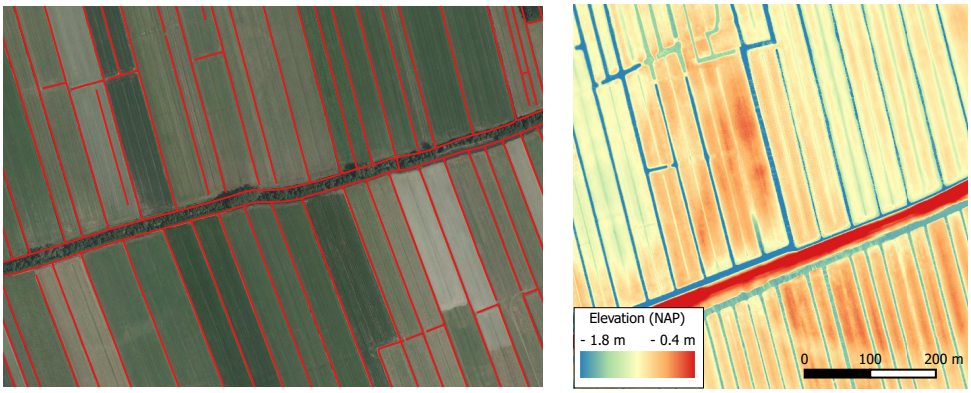
[a] *3D geoinformation, Delft University of Technology, Julianalaan 134, 2628BX Delft, The Netherlands*

Abstract

Drainage networks play a crucial role in protecting land against floods. It is therefore important to have an accurate map of the watercourses that form the drainage network. Previous work on the automatic identification of watercourses was typically based on grids, focused on natural landscapes, and used mostly the slope and curvature of the terrain. We focus in this paper on areas that are characterised by low-lying, flat, and engineered landscapes; these are characteristic to the Netherlands for instance. We propose a new methodology to identify watercourses automatically from elevation data, it uses solely a raw classified LiDAR point cloud as input. We show that by computing *twice* a skeleton of the point cloud — once in 2D and once in 3D — and that by using the properties of the skeletons we can identify most of the watercourses. We have implemented our methodology and tested it for three different soil types around Utrecht, the Netherlands. We were able to detect 98% of the watercourses for one soil type, and around 75% for the worst case, when we compared to a reference dataset that was obtained semi-automatically.

Keywords: LiDAR, skeleton, watercourse, Medial Axis Transform

*Corresponding author
Email address: r.y.peters@tudelft.nl (Ravi Peters)



(a) (b)

Figure 1: A typical landscape in the Netherlands, for a region around Utrecht. **(a)** Watercourses identified in red over an aerial image. **(b)** Elevation obtained from aerial laser scanning.

1. Introduction

Several areas around the world, such as the Netherlands, are characterised by low lying, flat, and engineered agricultural lands. As shown in Figure 1a, the drainage network of these areas—which is artificial—consists of *connected linear features* such as channels, culverts, and reshaped gullies (Bailly et al., 2011); we refer to these hereafter as "watercourses". These form a network that transits water from the fields into larger canals (Bouldin et al., 2004). Typically, these areas have very little variation in elevation, see in Figure 1b how the elevation varies only by around 1.4m over the area (0.5km^2). Because engineered lands are sensitive to flooding (Parry et al., 2007), it is of the utmost importance to have an up-to-date and accurate model of the watercourses (Cavalli et al., 2013). Such a model will consist of the planimetric geometry of the watercourses (their centreline), their connectivity, but also of other characteristics such as the width and the shape of the banks (which is useful to calculate the storage capacity). This information can help us design measures to avoid floods (Cazorzi et al., 2013), and can play an important role in designing drainage channels and pumping stations (Malano and Hofwegen, 1999).

In this paper, we investigate how the network of watercourses in flat and engineered landscapes can be *automatically* identified. Currently, such networks are typically identified with a semi-automatic methodology using Li-

DAR point clouds, aerial imagery and field surveys. This is labour-intensive, and subjective (Gandolfi and Bischetti, 1997). The vast majority of methods and algorithms developed in interdisciplinary studies have not been designed for our case, but for natural landscapes. These usually assume that the slope along a watercourse is always positive (Costa-Cabral and Burges, 1994; Lohani and Mason, 2001), or that the curvature of the terrain is higher than a certain threshold (Meisels et al., 1995; Brzank et al., 2005; Passalacqua et al., 2010). Furthermore, when LiDAR datasets are used, usually a derived product of the original dataset is used as input, e.g. a 5mX5m gridded digital elevation model (DEM) seems standard. This is an inherent problem since they contain missing data where the water is located (due to the absorption of LiDAR signals by water), and because the conversion to grids inevitably implies a certain decrease of accuracy, due to the interpolation process and the resampling (Gold and Edwards, 1992; Fisher, 1997; Brzank et al., 2008). Furthermore, as can be seen in Section 4, a 5m resolution DEM is not useful for our case because typically watercourses can be less than 1m wide, and several ones can be closer than 5m to each other. We further describe in Section 2 the few methodologies that have been designed for our case.

We present in Section 3 an automatic identification methodology that uses only a raw LiDAR dataset as input. The only requirement is that the elevation samples must be classified (into ground, water, buildings, and vegetation); algorithms for such classifications are readily available, see for instance Tóvári and Pfeifer (2005). We use two complementary skeleton-based methods — one in 2D and one in 3D — to extract the centrelines of the watercourses. The 2D method makes use of the absence of LiDAR points on water surfaces and constructs the alpha-shape of the LiDAR samples projected to the xy plane, then computes a 2D skeleton for the resulting polygons to obtain the centrelines. For watercourses that are dry and/or covered by canopy, we directly compute a 3D skeleton of the LiDAR points to identify concave profiles in the landscape and extract centrelines. We have implemented our methodology and we report in Section 4 on experiments in which we compare the results of our automatic methods with an existing watercourse dataset that was obtained in a semi-automatic manner. Our three study areas are located in rural areas around the city of Utrecht in the Netherlands, and differ in their type of subsoil (clay, peat, and sand). By combining both the 2D and 3D methods, we have been able to identify over 95% of the watercourses in the peat and clay areas, and around 75% in the sand area.

2. Related work

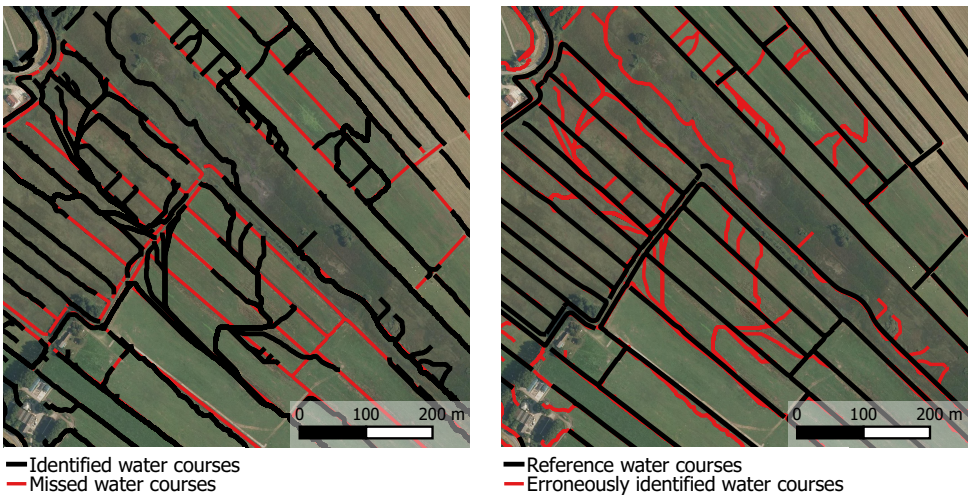
Bailly et al. (2008) identifies narrow watercourses in agricultural areas with slopes using LiDAR by analysing the profile at defined locations perpendicular to field boundaries, and choosing a threshold for the curvatures (those above the threshold are ditches). They achieved ditch omissions of around 50%, and ditch commissions of around 15%. They attribute the poor performance to insufficient sampling of LiDAR points (around 10 points/m^2), and to vegetation coverage along the ditches. Their method only works if the boundaries of fields are given as input, and may not perform well for watercourses which are largely filled with water.

Passalacqua et al. (2012) argue that watercourses can typically be characterised by positive curvature, and by high values of flow accumulation. Their method was designed specifically for flat and engineered landscapes. They successfully extracted the network using a 3m DEM for the low-relief human-impacted landscape of an area along a basin in the USA. However, their study area has elevation differences of up to 60m, and therefore seems to be less flat than the area we use for our study (see Figure 1b and Section 4). Since their method is freely available in the package GeoNet (Sangireddy et al., 2016), we have tested it for our area. Figure 2 shows that it performs poorly for our peat area (many errors of omission and commission), although it performed slightly better for the clay area. It struggles in places with very low relief since there is little surface curvature, and thus picks the slightest change.

Cazorzi et al. (2013) extracts local low-relief features from a 1m DEM, and extract the network by labelling peak values based on a threshold value that is taken as the standard deviation of the local relief. Their results proved to be more reliable than their outdated cartography-based reference data, and a median distance of reference points to the extracted watercourse network was registered to be about 1m. The usage of a threshold on the local relief can have implications on the ability of the method to identify watercourses of different forms, and as stated above, is less suitable for low-relief watercourses.

Cho et al. (2011) detect stream channels in very low-relief landscapes, based on local minima and maxima in elevation values from a 1m DEM, but comment that the method requires significant training and computation.

Possel et al. (2010) try to detect very wide (100m) buried channels in an area in the Netherlands from a 2m DEM with a maximum likelihood classifier



(a) Error of omission (b) Error of commission

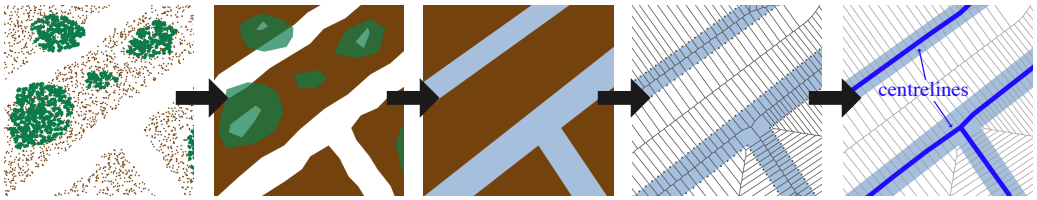
Figure 2: Identification of watercourses by Passalacqua et al. (2012) (with GeoNet software) for an area with peat soil near Utrecht in the Netherlands. The dataset is compared to a reference dataset provided by the HDSR (background aerial photo courtesy of `www.pdok.nl`).

based on slope, curvature and relative elevation.

Höfle et al. (2009) extract the edges of a water body by modelling the locations of laser shot dropouts along with the surface roughness, after which potential water regions are detected by using a region growing algorithm. Toscano et al. (2014) proposes a similar method that requires less pre-processing, but the method uses a DEM. The original LiDAR samples are converted (pixels having no LiDAR signal get a low value) and then an analysis of the height histogram allows them to identify low area (which should be water). We believe that histogram analysis will not suffice for smaller water bodies such as ours, since these do not generate high enough peaks in the elevation data. Both Höfle et al. (2009) and Toscano et al. (2014) are unable to classify dry watercourses or those completed covered by canopy.

3. Our methodology based on the 2D and 3D skeletons

We propose two skeleton-based approaches to automatically identify watercourses from a classified aerial LiDAR point cloud. The first one (see Figure 3a) uses the alpha-shape (also commonly called "concave hull") of



1. Ground + vegetation points projected to the ground plane.

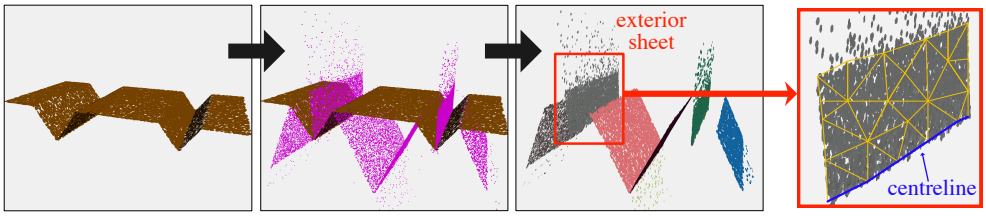
2. alpha-shapes of points. Use vegetation polygons to fill holes.

3. Construct water polygons from voids between simplified ground polygons.

3. Voronoi diagram from points on densified water boundary.

4. Select VD edges inside water polygon, prune and simplify to find the centrelines.

(a) Workflow to obtain watercourses from the 2D skeleton.



1. Ground points.

2. 3D skeleton of the ground points.

3. Segmentation of 3D skeleton in sheets.

4. Triangulate exterior sheets and select lower envelope to find the centrelines.

(b) Workflow to obtain watercourses from the 3D skeleton.

Figure 3: Our workflows that summarise how we extract the watercourses using 2D and 3D skeletons on a LiDAR point cloud.

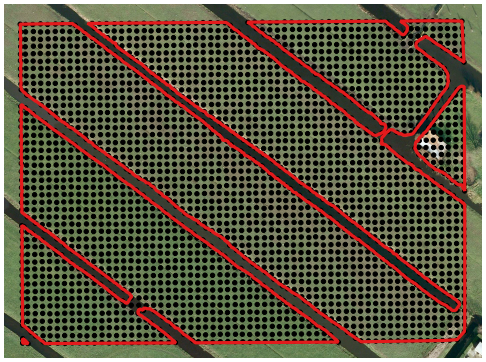
ground and vegetation points to compute water polygons from which the centrelines are derived using a 2D skeleton. The second one (see Figure 3b) computes centrelines as the lower envelope of the 3D skeleton of ground points and aims at identifying concave profiles, which means it can also identify dry watercourses and watercourses covered partly by canopy. Both 2D and 3D skeletons are used because they allow us to make use of different characteristics of the input point clouds, as further explained below.

Since the 2D method uses mostly a series of known algorithms and standard GIS operations, we shortly describe it in Section 3.1. We focus in Section 3.2 on the 3D skeleton-based method because it is most novel. Furthermore, since both methods have complementary strengths, we also describe how to integrate them into a unified and more robust approach that yields better overall results. For both methods we assume, as is the case in the Netherlands, to have a point cloud available that is classified into at least 1) ground points, 2) building points, 3) water points and 4) other points (e.g. vegetation).

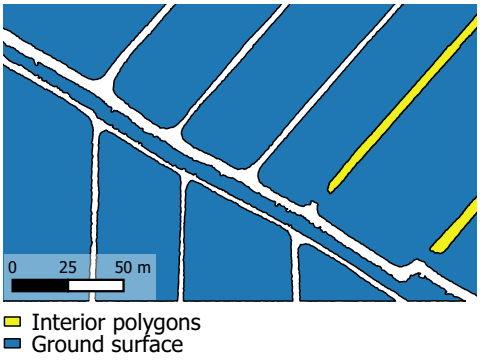
3.1. Computing centrelines using the 2D skeleton

We take advantage of a key property of red laser-based LiDAR datasets above open water bodies: it is almost completely absorbed, only LiDAR signals emitted at or near nadir are reflected strong enough to be detected by the sensor. The few LiDAR measurements which did reflect on the water bodies can be filtered out (Höfle et al., 2009). As input for this method we use two sets of points: 1) ground points (which includes the building points to fill the voids in the ground class where buildings are), 2) vegetation points. What remains is a dataset with separate disconnected groups of ground points, with voids in between these groups representing the waterbodies (see Figure 4). These groups of ground points are then converted into multiple disconnected ground polygons using *alpha-shapes* (see Edelsbrunner et al. (1983)); the alpha parameter was manually chosen based on the density of the LiDAR dataset.

We also consider vegetation points. Generally these should be omitted since they might cover waterbodies. However, there may also be vegetation that is so dense that it prevents LiDAR pulses from reaching the ground surface below. This results in holes —we call these *vegetation artefacts*— in the ground polygons as illustrated in Figure 5a. This is undesirable, since the alpha-shape method is based on the assumption that all voids in the point cloud are watercourses, which they are not in this case. To fix these artefacts,



(a) Displayed is a 3m uniformly spaced subset of LiDAR points, but alpha-shape was computed for the full set of points. Aerial photo as backdrop.



(b) The majority of the watercourses are represented by the space between these alpha shapes, while some are interior polygons of the alpha shapes.

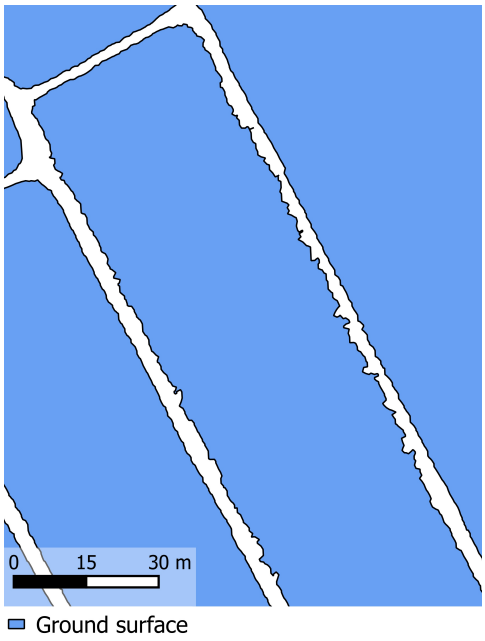
Figure 4: The alpha-shape of LiDAR points classified as ground.

we use vegetation polygons that are computed from the alpha shape of the vegetation points. We consider holes in the ground polygons that are fully covered by vegetation polygons to be vegetation artefacts and subsequently remove them. The resulting ground polygons are then free of vegetation artefacts. Polygons representing the watercourses can now be obtained by taking the inverse of the ground polygons (see step 3 in Figure 3a). Remaining irregularities in the water polygons are removed using a "buffer-debuffer operation" to fill tiny holes, followed by a Douglas-Peucker line simplification (Douglas and Peucker, 1973) to smooth the boundary (see Figure 6a).

Finally, we compute the Voronoi diagram (VD) of the boundaries of the water polygons. We do this by first discretising the polygon boundaries (e.g. with point every 1m). Then we compute the VD of the resulting set of points, as illustrated in Figure 7; notice that we do not compute the VD of line segments (Held, 2001). The 2D skeleton is a subset of this VD, it is defined as the set of edges in the VD of the polygon that completely reside in the polygon's interior, which is conceptually similar to the algorithm described in Gold and Snoeyink (2001).

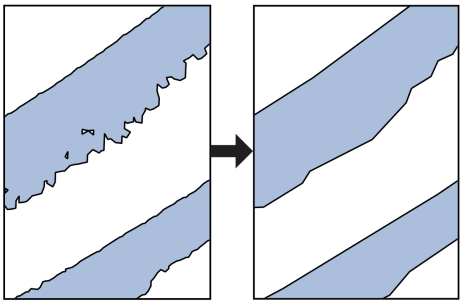
When using the VD of a polygon boundary to compute the 2D skeleton, a number of *unwanted* branches are generated next to the desired centrelines;

(a) The alpha shapes of the vegetation points cover the vegetation artefacts.

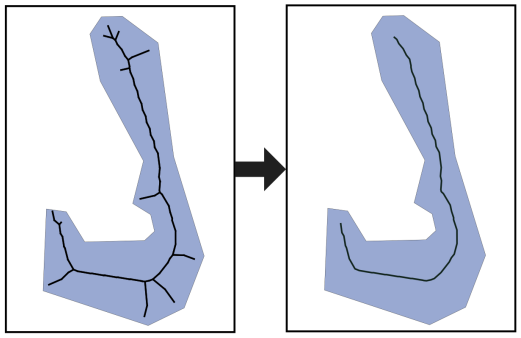


(b) A clean dataset of ground surface alpha shapes is formed by removing the vegetation artefacts.

Figure 5: Vegetation artefacts are removed by detecting vegetation polygons that are completely surrounded by ground.



(a) Watercourses are regularized using buffer-debuffering and Douglas-Peucker line simplification.



(b) centrelines are regularised using pruning.

Figure 6: Regularisation of polygon boundaries and skeleton results for the 2D skeleton based method.

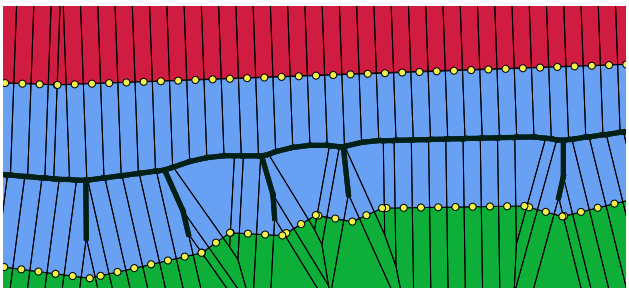


Figure 7: The VD of a set of densified boundary points is used to compute the 2D skeleton (black) of a water polygon (blue), in between two river banks (red and green). The 2D skeleton is defined as all edges of the VD which are completely within the water polygons.

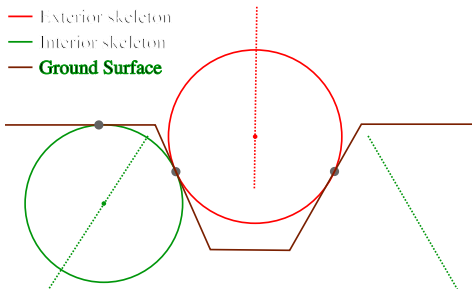
this is a normal property of the skeleton. These unwanted branches are defined as any edge incident to a vertex of degree 3 and one of degree 1. We automatically prune these unwanted branches by setting a threshold on the branch length (Figure 6b).

3.2. Computing centrelines using the 3D skeleton

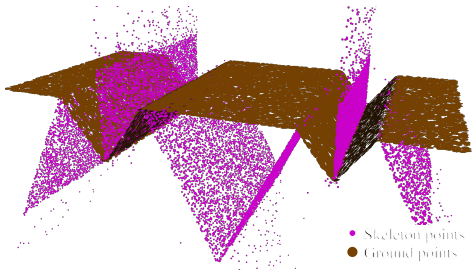
Whereas the 2D skeleton method is based on the absorption of the red LiDAR signal by water surfaces, the 3D skeleton method is based on the three-dimensional morphology of the landscape to identify watercourses and allows us to identify watercourses that are dry or covered by canopy.

Our 3D skeleton approach is based on the Medial Axis Transform that was originally introduced by Blum (1967); the MAT — called 3D skeleton in the remainder of this text — models the space between surfaces as a collection of medial balls. Each medial ball is tangent to the surface at two or more points and the centres of the medial balls form a medial skeletal structure of the object (see Figure 8a). Similar to the 2D skeleton there are branches structured in a hierarchy, but in the 3D skeleton these are surfaces that we call *medial sheets* (see Figure 8b). The 3D skeleton of a typical watercourse results in three medial sheets: one exterior (above ground), and two interior (under ground). The exterior medial sheet of a watercourse thus forms a 'centre plane' that contains the centreline of the watercourse. We define the centreline of a watercourse as the projection to the xy-plane of the lower envelope of its exterior medial sheet (see also Figure 3b).

For the computation of the 3D skeleton we use the shrinking ball algorithm of Ma et al. (2012), extended with the denoising heuristics from Peters and Ledoux (2016). We compute the 3D skeleton for all points that are clas-



(a) Cross section of a watercourse. The 3D skeleton is obtained by reconstructing medial balls that touch the ground surface in two points.



(b) Perspective view of the 3D skeleton of simple watercourses.

Figure 8: Profile view of 3D skeleton of the terrain

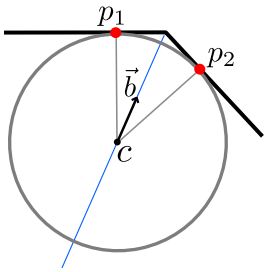
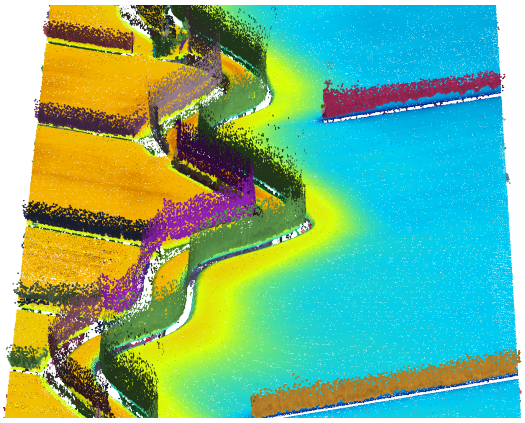
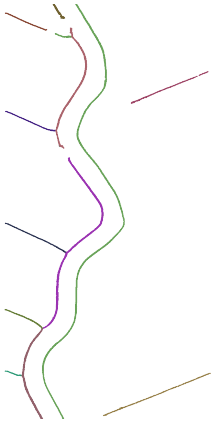


Figure 9: Each medial ball touches the surface in two points (p_1 and p_2). The medial bisector $\vec{b}$ is defined as the bisector of the vectors from the center c to p_1 and p_2.

sified as ground. The result is a point approximation (i.e. a point cloud) of the 3D skeleton. Next, we perform a segmentation of the point cloud into distinct medial sheets using a region-growing segmentation algorithm, conceptually similar to the one described in Rabbani et al. (2006), but instead of the angular difference in normals we use the angular difference in medial bisector. The medial bisector (as defined in Figure 9) is very suitable to distinguish between different medial sheet because it is by definition locally similar within the same sheet, but different where two sheets intersect (see Siddiqi and Pizer (2008) for more details). An added benefit of this approach is that outliers in the 3D skeleton point cloud can be omitted, since they are not part of any medial sheets and are therefore not assigned to a segment. Figure 10 shows the segmentation result for a LiDAR dataset of watercourses. Observe that each watercourse is delineated by a medial sheet.



(a) Perspective view of skeleton and ground points.



(b) Plan view of exterior skeleton sheets.

Figure 10: Segmentation of 3D skeleton sheets. Each distinct sheet was assigned a random colour. The surface points are coloured by elevation (yellow = low; blue = high).

Prior to deriving the 2D centreline representation of each sheet, we triangulate the medial sheets using the ball pivoting algorithm from Bernardini et al. (1999). A centreline of the water surface can then be derived by selecting the lower edge segments on the boundary of the triangulated sheet (see also step 4 in Figure 3b). We identify these by walking around the boundary edges of the triangulation (i.e. those edges that are only incident to one triangle), and then selecting the edges whose two endpoints are below (i.e. having a lower z coordinate) the opposite vertex in the triangle to which that edge belongs. We finally obtain a 2D representation of the resulting polyline simply by omitting the z-coordinates.

3.3. Combined approach

In Section 4 we show that the 2D and 3D skeleton methods have different strength. We are able to combine the strengths of both methods using a few common GIS operations. First we buffer the resulting centrelines from both methods. If we then dissolve these buffers, one polygon remains for every watercourse. Finally, the centrelines of these polygons are generated using the 2D skeleton again (see Figure 11).

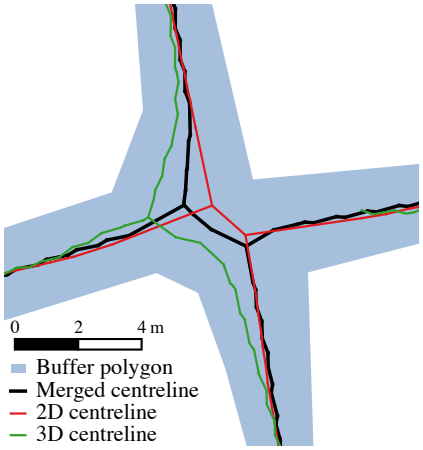


Figure 11: The centreline generated by the 2D and 3D skeleton methods are combined by merging the centrelines, subsequently buffering them, and lastly by generating a new centreline using the 2D skeleton centreline extraction method.

4. Experiments & results

We have implemented the methods described in the previous section using a combination of existing tools and our own programs. The 2D skeleton method was implemented using primarily QGIS[1] and LAStools[2]. For the 3D skeleton method, we have used primarily our own software implementations[3], with the exception of the ball pivoting algorithm for which we used Meshlab[4].

4.1. Study area & experiments

Our study areas are all 3x3 km and are situated around the city of Utrecht, the Netherlands (see Figure 12). This area consists for the most part of flat (elevation typically ranges between -2 m to +6 m) and engineered landscapes (see Figure 13). We have selected three different types of environments with different characteristics that can be classified according to their subsoils; clay, peat, and sand:

Clay: little vegetation and fairly wide watercourses with a very clear concave profile.

[1]QGIS: `http://www.qgis.org`.
[2]LAStools: `https://rapidlasso.com/lastools/`.
[3]partially open source at `https://github.com/tudelft3d/masbcpp`
[4]MeshLab: `http://meshlab.sourceforge.net`.

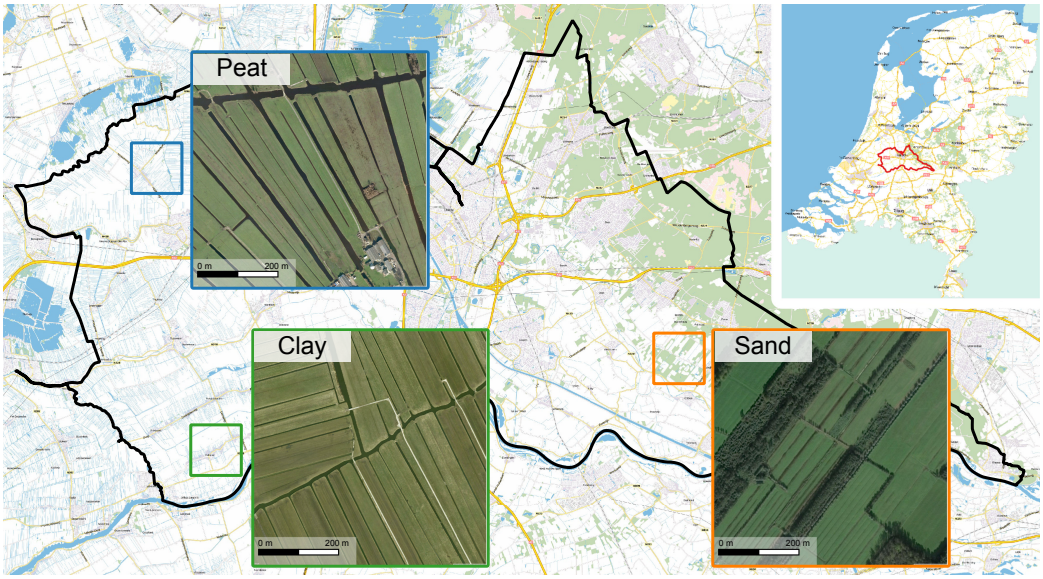


Figure 12: The three study areas we use in our experiments.

Peat: little vegetation and very wide watercourses with a less clear concave profile.

Sand: a lot of vegetation and narrow watercourses with a clear concave profile.

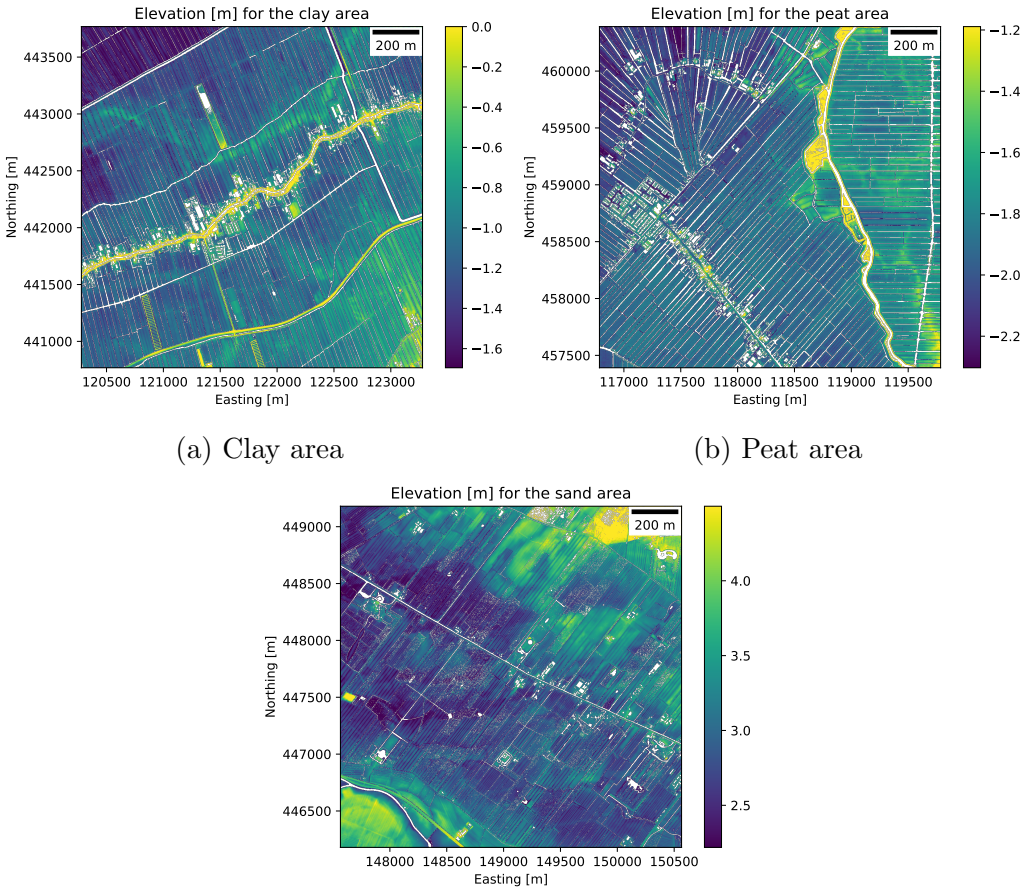
More detailed information on the study areas can be found in Table 1.

The publicly available AHN3[5] aerial LiDAR point cloud data was used, which is the most current version of the national elevation dataset of the Netherlands, it has a point density of around 10 points per square meter. For validation purposes, we have used an existing centreline dataset, which was obtained with a semi-automatic method, from HDSR (Hoogheemraadschap De Stichtse Rijnlanden, i.e. the water board responsible for water management in our study areas).

For the experiments, first we applied our two skeleton-based approach separately on all three study areas. Then we combined the results using the method described in Section 3.3. The generated datasets of watercourses were compared to the HDSR dataset of watercourses, i.e. our reference dataset. We use the following error metrics (see Lillesand et al. (2008)):

- *Positional accuracy*: Refers to the extent to which the actual position

[5] `www.ahn.nl`



(a) Clay area

(b) Peat area

(c) Sand area

Figure 13: Terrain elevation of the study areas. Interpolated from ground points. No data pixels are white.

Table 1: Details on study areas. The specified location in EPSG:28992 is the lower left coordinate of the area (they are all 3x3 km in size). The percentage of vegetation coverage is based on the relative number of vegetation points in the point cloud. The percentage of water coverage is computed by taking the total surface area of all water polygons in the HDSR reference dataset and dividing it by the total surface area of the study area.

	Area		
Characteristic	Clay	Peat	Sand
Location (EPSG:28992)	(120279 , 440768)	(116785 , 457391)	(147565 , 446180)
Location (city / village)	Cabauw	Zegveld	Langbroek
Vegetation coverage (%)	5	8	47
Water coverage (%)	9	14	5
Elevation range (cm)	-250/+300	-250/+150	+150/+600

of the watercourses is correctly indicated. It can be estimated by calculating the average positional deviation for multiple watercourses in the generated dataset with respect to the reference dataset.

- *Error of omission*: The percentage of watercourses in the reference dataset that are *not* in the generated dataset.
- *Error of commission*: The percentage of watercourses in the generated dataset that are *not* in the reference dataset.

To compute the error metrics we have uniformly discretised the centrelines into points, and then identified for these points what is the shortest Euclidean distance to the centreline in the reference dataset. By aggregating and averaging these point distances per centreline, an estimate is obtained of the generated dataset's positional accuracy. To obtain the mapping accuracies we use threshold distances, e.g. if a threshold distance of 2 m is set, and the distance between a point on the generated centreline and the reference centreline is larger than this distance, then this point counts as an error of commission. Similarly, points can be selected on the generated centreline to find the error of omission. The metrics are computed by taking the number of points omitted or committed, relative to the total number of points.

Table 2: This table lists the metrics which were computed for the centrelines generated by the 2D skeleton, the 3D skeleton, and the combined approach, for clay, peat and sand areas.

		dataset		
Error metric		Clay	Peat	Sand
Positional accuracy (m)	2D skeleton	0.5	0.7	0.6
	3D skeleton	0.6	0.8	0.8
	Combined	0.6	0.7	0.9
Error of omission (%)	2D skeleton	5	5	58
	3D skeleton	4	15	26
	Combined	2	3	24
Error of commission (%)	2D skeleton	1	2	4
	3D skeleton	8	8	17
	Combined	8	8	17

Notice that our evaluation method is comparable to the one proposed by Heipke et al. (1997). The main difference is that Heipke et al. (1997) use a buffer to match line features between the reference and the generated data instead of the shortest distance between the points on the line features. A benefit of our approach is that it can do partial matching, i.e. a parts of the same line feature are matched separately, whereas Heipke et al. (1997) match only the complete line features. We believe this is an advantage espically in the case where the topology of the networks of the reference and the generated datasets are different, which is very likely since they are generated using different methods.

4.2. Results

The outcome of our experiments, for all combinations of methods and study areas, are shown in Table 2 and summarised below.

Both our skeleton-based methods perform very well for the clay area with

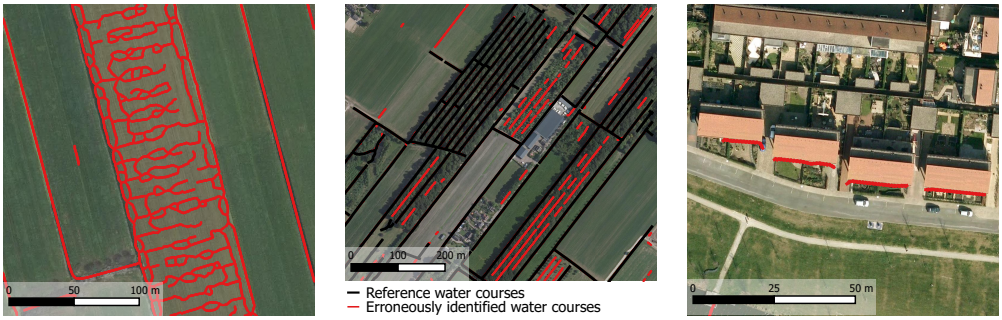
only 5% of watercourses missing (error of omission) and a low error of commission. As we explain in Section 3.1, our 2D skeleton method works based on the assumption that water surfaces can be detected from the point cloud because water typically is the only surface type that does not reflect red laser. The 3D skeleton method on the other hand works based on the assumption of concavity, i.e. watercourses often have significant surface curvature at the water banks. It is therefore not surprising that both methods perform well on the clay area that has significant concavity and wide watercourse surfaces with little vegetation covering it.

For the peat area, the 2D skeleton performed nearly equally well, since water surfaces are even wider and again there is little vegetation covering these surfaces. However, the watercourses display less clear concave profiles, since relative water levels are higher here (thus many watercourses have only very small banks), which impedes the effectiveness of the 3D skeleton. Although the 3D skeleton performs less for this area, it still manages to identify some of the watercourses which were not identified by the 2D skeleton method. This is indicated by the fact that the combined 2D+3D skeleton method identifies roughly 97% of the watercourses, which is more than the 2D skeleton method identified by itself.

The sand area clearly stands out in Table 2 since the errors for both methods are significantly worse than for the two other areas. Especially the 2D skeleton method does a poor job at identifying the watercourses with a 58% error of omission. The main problems here are: (1) the fact that water is not well visible in this landscape, (2) water surfaces are often narrower than 1m, and (3) many patches of forest are present. The 3D skeleton is much more effective with only 26% error of omission, but it also struggles with the identification of the narrower watercourses that are naturally also represented with relatively few points in the point cloud. The combined method for the sand area raises the commission error only marginally to 24%, indicating that the 3D skeleton method identified almost all of the watercourses identified by the 2D skeleton, and is clearly the better performing method for this area.

5. Discussion

Our 2D- and 3D-skeleton methods both have different strengths and limitations. Summarising, we can say that the 2D skeleton method is particularly efficient with open water watercourses of sufficient width. The (lack of) surface curvature does not affect its effectiveness. And it is characterised by



(a) Artefact centrelines in the 3D skeleton caused by local relief.

(b) Error of commission of the sand area for the 3D skeleton.

(c) Missing data due to shadowing causes false centrelines for the 2D skeleton.

Figure 14: Remarkable cases in the results of our experiments.

a low error of commission. Its limitations are watercourses with a width of less than 1 m (i.e. depending on the parameters used while creating the alpha-shapes, which itself depends on the density of the LiDAR point cloud; for a denser point cloud, this parameter could be lowered significantly), the presence of large patches of vegetation that cover water bodies (due to the cleaning method of vegetation artefacts), and voids in the LiDAR point cloud that are not the result of waterbodies. The former also means that the 2D skeleton method is ineffective in case of low water levels at time of LiDAR measurements. Thus, the 2D skeleton method is particularly suited for use in areas where water levels are high and water is a predominant feature of the landscape. The 3D skeleton on the other hand does not depend on the presence of water or voids in general, and is effective as long as the canopy is not too dense and allows the LiDAR signal to pass through. The limitations of the 3D skeleton are its dependence on surface curvature (high water levels may obfuscate this) an its tendency to find concavities in the landscape where one would not immediately expect a watercourse, e.g. levees or piles of earth or dirt (see Figure 14a).

The 3D skeleton method is therefore mostly suited for areas with watercourses that have a low water level, clearly concave profiles and may be covered with large patches of vegetation.

Our approach to combine the results of both the skeleton-based method is effective because it always decreases the commission and omission errors.

Only the error of commission never improves since watercourses are never discarded by our current combination approach.

Finally, we note that the reference dataset that we used is not a perfect ground truth: we discovered many errors and this should be considered when evaluating the error metrics, Multiple situations were encountered where a watercourse was present in the reference dataset, which was clearly not present in the point cloud data and the resulting generated datasets. The opposite happened especially in the sand area where our 3D skeleton method in particular identified a large number of watercourses that are not present in the reference dataset, but which based on shape and size clearly were valid watercourses. These watercourses are hard to identify, since they are covered by forest, thus this likely indicates an incompleteness of the reference dataset. While the 3D skeleton shows an error of commission of over 17% for the sand area, we estimate that at least half of these are not actually errors, but rather watercourses that were missing in the reference dataset, and do exist in reality (see Figure 14b). We also noticed some erroneous results due to deficiencies of the point cloud, such as shadowing effects. In case of the 2D skeleton method this may lead to the creation of voids that are not watercourses, but would still be identified as such. This is especially the case near buildings (see Figure 14c). As a last remark we note that especially the 2D skeleton method very much depends on a correct classification of the point cloud into the necessary classes (ground, vegetation, buildings and water).

6. Conclusions and future work

With omission rates of only 2%, for one soil type, we have successfully demonstrated that we are able to automatically derive centrelines for watercourse from classified aerial point clouds in the Netherlands. While the effectiveness of our 2D+3D skeleton-based approach depends on various properties of the landscape (such as vegetation coverage and curvature), the same can be said of current approaches, as we have noticed from the authoritative reference dataset (obtained using a semi-manual approach) that we used to validate our results. Furthermore, by combining the results of both 2D and 3D skeletons we are able to compensate for this to some degree. By using highly accurate raw point cloud data as input, we are able to identify very narrow watercourses of less than 1m wide, whereas existing automated methods from literature are typically limited to the cell size of a grid (often 5mX5m). Thus, the strengths of our skeleton-based approach are that it is

automatic, effective, that is it able take full advantage of a raw point cloud dataset (instead of using a derivative), and it is feasible to implement.

There are a number of things we believe deserve further research. First, and most interestingly, the computation of additional watercourse properties such as the width and shape of the banks, and the water storage capacity. We believe that especially our novel 3D skeleton method has great potential here, since it has a well-defined link to the watercourse banks, and also defines a volume through its medial balls (which fill the watercourse). Second, to examine how our methods performs on an area with significant relief. And finally, to make our methodology work with larger areas, since in this work we have only looked at relatively small study areas. We are at this moment limited by the main memory of a computer, but it should be trivial to scale the computation of the 2D skeleton using a tiling scheme. Futhermore, Peters and Ledoux (2016) showed that also the 3D skeleton can be constructed for billions of points using a tiling scheme.

Acknowledgements

We would like to express our gratitude to *Hoogheemraadschap De Stichtse Rijnlanden* (HDSR) for providing us with useful insights and data. This research is supported by the Dutch Technology Foundation STW, which is part of the Netherlands Organisation for Scientific Research (NWO), and which is partly funded by the Ministry of Economic Affairs (project code: 12217).

References

Bailly, J. S., Lagacherie, P., Millier, C., Puech, C., Kosuth, P., 2008. Agrarian landscapes linear features detection from lidar: application to artificial drainage networks. International Journal of Remote Sensing 29 (12), 3489–3508.

Bailly, J. S., Levavasseur, F., Lagacherie, P., 2011. A spatial stochastic algorithm to reconstruct artificial drainage networks from incomplete network delineations. International Journal of Applied Earth Observation and Geoinformation 13 (6), 853–862.

Bernardini, F., Mittleman, J., Rishmeier, H., Silva, C., Taubin, G., Oct 1999. The ball-pivoting algorithm for surface reconstruction. Visualization and Computer Graphics, IEEE Transactions on 5 (4), 349–359.

Blum, H., 1967. A transformation for extracting new descriptors of shape. Models for the perception of speech and visual form 19 (5), 362–380.

Bouldin, J., Farris, J., Moore, M., Cooper, C., 2004. Vegetative and structural characteristics of agricultural drainages in the mississippi delta landscapes. Environmental Pollution 132 (3), 403 – 411.

Brzank, A., Göpfert, J., Lohmann, P., 2005. Aspects of lidar processing in coastal areas. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (Part 1/W3), 6.

Brzank, A., Heipke, C., Goepfert, J., Soergel, U., 2008. Aspects of generating precise digital terrain models in the wadden sea from lidar-water classification and structure line extraction. ISPRS Journal of Photogrammetry and Remote Sensing 63 (5), 510–528.

Cavalli, M., Trevisani, S., Goldin, B., Mion, E., Crema, S., Valentinotti, R., 2013. Semi-automatic derivation of channel network from a high-resolution dtm: the example of an italian alpine region. European Journal of Remote Sensing 46, 152–174.

Cazorzi, F., Fontana, G. D., De Luca, A., Sofia, G., Tarolli, P., 2013. Drainage network detection and assessment of network storage capacity in agrarian landscapes. Hydrological Processes 27, 541–553.

Cho, H., Slatton, K. C., Krekeler, C. R., Cheung, S., 2011. Morphology-based approaches for detecting stream channels from alsm data. International Journal of Remote Sensing 32 (24), 9571–9597.

Costa-Cabral, M. C., Burges, S. J., 1994. Digital elevation model nnetwork (demon): A model of flow over hillslopes for computation of contributing and dispersal areas. Water Resources Research 30 (6), 1681–1692.

Douglas, D., Peucker, T., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. The Canadian Cartographer 10 (2), 112–122.

Edelsbrunner, H., Kirkpatrick, D., Seidel, R., 1983. On the shape of a set of points in the plane. IEEE Transactions on Information Theory IT-29 (4), 551–558.

Fisher, P. F., 1997. The pixel: A snare and a delusion. International Journal of Remote Sensing 18 (3), 679–685.

Gandolfi, C., Bischetti, G. B., 1997. Influence of the drainage network identification method on geomorphological properties and hydrological response. Hydrological Processes 11, 353–375.

Gold, C. M., Edwards, G., 1992. The Voronoi spatial model: Two- and three-dimensional applications in image analysis. ITC Journal 1, 11–19.

Gold, C. M., Snoeyink, J., 2001. A one-step crust and skeleton extraction algorithm. Algorithmica 30, 144–163.

Heipke, C., Mayer, H., Wiedemann, C., Jamet, O., 1997. Evaluation of automatic road extraction. International Archives of Photogrammetry and Remote Sensing 32 (3 SECT 4W2), 151–160.

Held, M., 2001. vroni: An engineering approach to the reliable and efficient computation of Voronoi diagrams of points and line segments. Computational Geometry—Theory and Applications 18 (2), 95–123.

Höfle, B., Vetter, M., Pfeifer, N., Mandlburger, G., Stötter, J., 2009. Water surface mapping from airborne laser scanning using signal intensity and elevation data. Earth Surface Processes and Landforms 34, 1635–1649.

Lillesand, T., Kiefer, R., Chipman, J., 2008. Remote Sensing and Image Interpretation, 6th Edition. John Wiley & Sons, Ltd.

Lohani, B., Mason, D. C., 2001. Application of airborne scanning laser altimetry to the study of tidal channel geomorphology. ISPRS Journal of Photogrammetry and Remote Sensing 56 (2), 100–120.

Ma, J., Bae, S., Choi, S., 2012. 3d medial axis point approximation using nearest neighbors and the normal field. The Visual Computer 28 (1), 7–19.

Malano, H., Hofwegen, P. V., January 1999. Management of Irrigation and Drainage Systems: A Service Approach. A. A. Balkema Publishers, Rotterdam, Netherlands.

Meisels, A., Raizman, S., Karnieli, A., 1995. Skeletonizing a dem into a drainage network. Computers & Geosciences 21 (1), 187–196.

Parry, M., Canziani, O., Palutikof, J., van der Linden, P., Hanson, C., 2007. Contribution of working group ii to the fourth assessment report of the intergovernmental panel on climate change. IPCC Fourth Assessment Report: Climate Change 2007.

Passalacqua, P., Belmont, P., Foufoula-Georgiou, E., 2012. Automatic geomorphic feature extraction from lidar in flat and engineered landscapes. Water Resources Research 48 (3).

Passalacqua, P., Do Trung, T., Foufoula-Georgiou, E., Sapiro, G., Dietrich, W. E., 2010. A geometric framework for channel network extraction from lidar: Nonlinear diffusion and geodesic paths. Journal of Geophysical Research: Earth Surface 115 (F1).

Peters, R., Ledoux, H., 2016. Robust approximation of the medial axis transform of lidar point clouds as a tool for visualisation. Computers & Geosciences 90 (Part A), 123–133.

Possel, B. M., Lindenbergh, R. C., Storms, J. E., 2010. Automatic detection of buried channel deposits from dense laser altimetry data.

Rabbani, T., Van Den Heuvel, F., Vosselmann, G., 2006. Segmentation of point clouds using smoothness constraint. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (5), 248–253.

Sangireddy, H., Stark, C. P., Kladzyk, A., Passalacqua, P., 2016. GeoNet: An open source software for the automatic and objective extraction of channel heads, channel network, and channel morphology from high resolution topography data. Environmental Modelling & Software 83, 58–73.

Siddiqi, K., Pizer, S., 2008. Medial Representations: Mathematics, Algorithms and Applications. Vol. 37. Springer, New York, NY, USA.

Toscano, G., Gopalam, U., Devarajan, V., 2014. Auto hydro break line generation using LiDAR elevation and intensity data. In: Proceedings ASPRS 2014 Annual Conference. Louisville, Kentucky, USA.

Tóvári, D., Pfeifer, N., 2005. Segmentation based robust interpolation-a new approach to laser data filtering. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36 (3/W19), 79–84.