GIS TECHNOLOGY

Feature aware Digital Surface Model analysis and generalization based on the 3D Medial Axis Transform

PhD Research Proposal

R.Y. Peters, MSc

GISt Report No. 65

OTB Research Institute for Housing, Urban and Mobility Studies

Delft University of Technology

Feature aware Digital Surface Model analysis and generalization based on the 3D Medial Axis Transform

PhD Research Proposal

R.Y. Peters, MSc

GISt Report No. 65

Summary

Modern Digital Surface Models (DSMs) are highly detailed and cover large areas. This brings great advantages for applications such as flood modeling, crisis management and 3D city modeling. Unfortunately, and despite recent developments on this subject, current methods are unable to fully take advantage of modern DSMs. First, because of their huge data volume. And second, because conventional methods use only 2.5D data-structures and algorithms. As a result of the latter, valuable 3D information that is present in modern DSMs is ignored.

This research aims to develop a methodology for the analysis and generalisation of modern DSMs that uses truly 3D data-structures and algorithms. It will be based on the Medial Axis Transform (MAT), a compact descriptor of the geometry and topology of shapes. It is the hypothesis of this research that the MAT can facilitate truly 3D analysis and generalization of modern 3D DSMs, through the definition of *features* (significant and application dependent surface characteristics) based on the geometrical and topological properties of the MAT.

The appropiate algorithms and data-structures will be designed, prototyped and robustly implemented. To support this process, a number of case studies will be performed, that each focuses on a distinct and practical application for which conventional 2.5D data-structures and algorithms have proven to be unsatisfactory.

ISBN: 978-94-6186-287-7
ISSN: 1569-0245

Section GIS technology
OTB Research Institute for the Built Environment
TU Delft
Jaffalaan 9, 2628 BX, Delft, The Netherlands
Tel.: +31 (0)15 278 4548; Fax +31 (0)15-278 2745
Websites: http://www.otb.tudelft.nl/
http://www.gdmc.nl/
Email: r.y.peters@tudelft.nl

Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Objectives . . . 4

2 Related work and background theory **5**
- 2.1 State of the art in Digital Surface Models . . . 5
  - 2.1.1 3D Surface Models . . . 6
  - 2.1.2 Finding and reconstructing features in the terrain . . . 7
  - 2.1.3 Terrain simplification . . . 9
  - 2.1.4 Dealing with huge datasets . . . 10
- 2.2 Shape analysis and matching . . . 11
- 2.3 The Medial Axis Transform . . . 13
  - 2.3.1 Defining the Medial Axis Transform . . . 14
  - 2.3.2 Properties and characteristics of the Medial Axis Transform . . . 15
  - 2.3.3 Computing the Medial Axis Transform . . . 16
  - 2.3.4 Pruning the Medial Axis Transform . . . 25

3 PhD research **31**
- 3.1 Research question . . . 31
- 3.2 General approach . . . 32
- 3.3 Methodology . . . 32
- 3.4 Validation . . . 35
- 3.5 Scope/Priorities . . . 35
- 3.6 Risks . . . 36
- 3.7 Case studies . . . 36
- 3.8 Contributions . . . 39

4 Practical aspects **41**
- 4.1 Technical matters . . . 41
- 4.2 Graduate School: courses and obligations . . . 42
- 4.3 Involved parties . . . 43

4.4 Planning 43

Bibliography **45**

Acronyms

DEM	Digital Elevation Model
DSM	Digital Surface Model
DTM	Digital Terrain Model
DT	Delaunay Triangulation
GIS	Geographic Information Science
MAT	Medial Axis Transform
OGC	Open Geospatial Consortium
TIN	Triangular Irregular Network
VD	Voronoi Diagram

Chapter 1

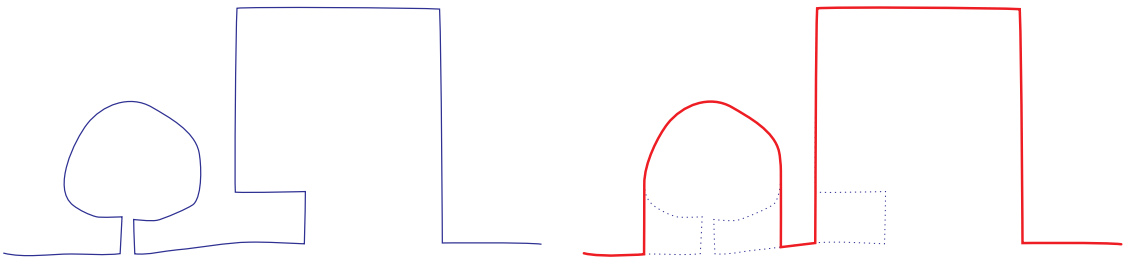
Introduction

1.1 Motivation

With recent advances in remote sensing technologies such as LiDAR, photogrammetry and multi-beam echo-sounding we are able to acquire geometric data of the Earth's surface in unprecedented quantity and accuracy. The resulting Digital Surface Models (Digital Surface Model (DSM)s) have a broad range of applications such as flood modeling, dike monitoring, crisis management and 3D city modeling [Snyder, 2013]. One of the fist LiDAR-based DSMs with national coverage was the Dutch *Actueel Hoogtebestand Nederland* (AHN). Its current iteration (AHN2) offers an average point density of 8 points per square meter, and peaks at several tens of points per square meter in urban areas, totaling up to hundreds of billions point measurements for the whole of the Netherlands [ahn]. Furthermore, AHN3 will be an even richer dataset with (at least) additional point attributes such as intensity, echo count and more classifications. Obviously, the wealth of data that results has big potential for the aforementioned applications.

However, it proves challenging to efficiently manage and process huge DSMs such as AHN2. Two not entirely unrelated issues can be identified here. First, because modern DSMs are so huge, they do not fit in a computer's memory [Isenburg et al., 2006a]. As a result, many of the conventional software tools are extremely inefficient with large datasets, or even refuse to finish processing entirely. And second, in GIS, a DSM is commonly treated as a 2.5D surface such as a raster or a TIN. While this alleviates memory requirements and simplifies computation, it comes at the price of a significant loss of information because 2.5D data structures are simply unable to completely represent data that is in fact 3D (see Figure 1.1). Consequently, the information that is lost during conversion from precise and fully 3D point clouds to (approximate) 2.5D surfaces, is no longer available for any subsequent processing. This issue is even more critical for non-aerial, i.e. mobile mapping, pointclouds.

The issue of handling huge DSMs with limited resources has attracted the attention of a number of researchers. Isenburg et al. [2006a] introduced the streaming paradigm for massive point clouds. They convincingly demonstrated that operations, such as rasterization or triangulation, that require only local information in the DSM, can be performed with great efficiency for huge DSMs on a regular laptop

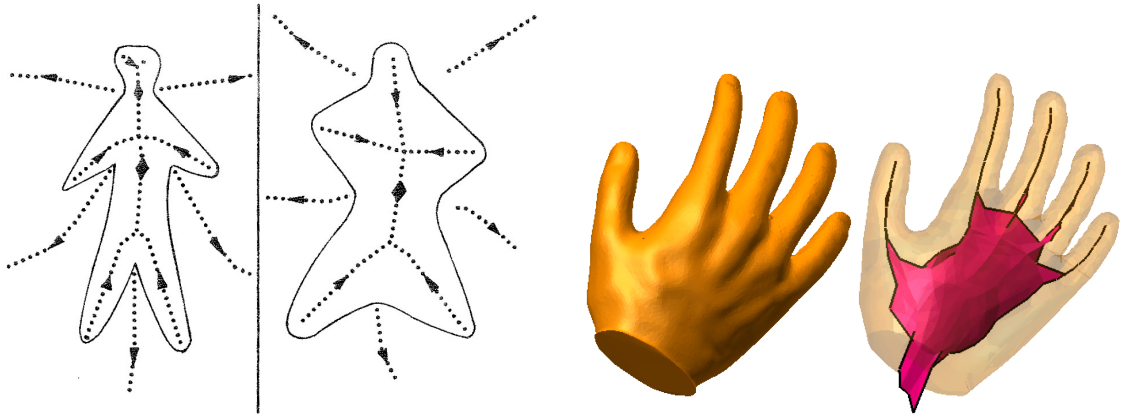


(a) Actual boundary (b) Boundary after 2.5D projection

Figure 1.1: Profile view of a DSM. Information is lost when the surface must be uniquely projectable to a horizontal plane.

computer. This technique is now widely used in industry and has been integrated in major GIS packages such as ESRI's ArcGIS. Other researchers have applied external memory techniques, for instance for the realtime visualisation of massive DSMs [Kreylos et al., 2008]. Furthermore, industry-backed resources are being invested in the development of new Open Geospatial Consortium (OGC) standards that facilitate the implementation of complete ICT infrastructures that handle the data management, dissemination, processing and visualization of massive point clouds [esc]. Thus, although software for *managing* huge DSMs may not be ubiquitous in practice just yet, the appropriate techniques do exist and are becoming increasingly available.

Nonetheless, the misuse of 2.5D data structures and algorithms for the processing of modern 3D DSMs persists. Let us, for example, look at *thinning*, a common DSM operation. Thinning reduces the number of elements (points, triangles or pixels, but I will talk about points from here on) that represent a DSM, preferably while preserving any defining surface charactistics (*features*) as well as possible. This may be one of the first steps in a DSM processing pipeline, since it will reduce the amount of CPU time (and thus money) required to execute the remaining part of the pipeline. Standard approaches implemented in Geographic Information Science (GIS) packages, such as ArcGIS, include random or n^{th} point selection, constraining the number of points contained by a moving rectangular window, and variants of TIN decimation (similar to Lee [1989]). The first two are, because of their simplicity, quite fast to execute, but they treat all points equally, because the chance of removal is equal for every point. Yet, some points are more valuable than others, that is particularly true for points that describe significant surface features, which in addition may be sparsely sampled. The thinning operator should therefore attempt to preserve these points, and prioritize the removal of points that are relatively unimportant (for example in large planar regions). Thinning based on TIN decimation actually attempts this, albeit from a 2.5D perspective. In TIN decimation, points are assigned an importance value based on the vertical variation of their local neighbourhood and the least important points are the most likely to be removed. Although this method is much more expensive than the earlier two, it gives much better results, typically removing more points while staying closer to the original surface [Crawford, 2013]. Unfortunately, as noted by Crawford [2013], this method should only be used for LiDAR based DSMs with 1^{st}-return surfaces (thus ignoring any following reflections for the same laser pulse due to penetrable surfaces



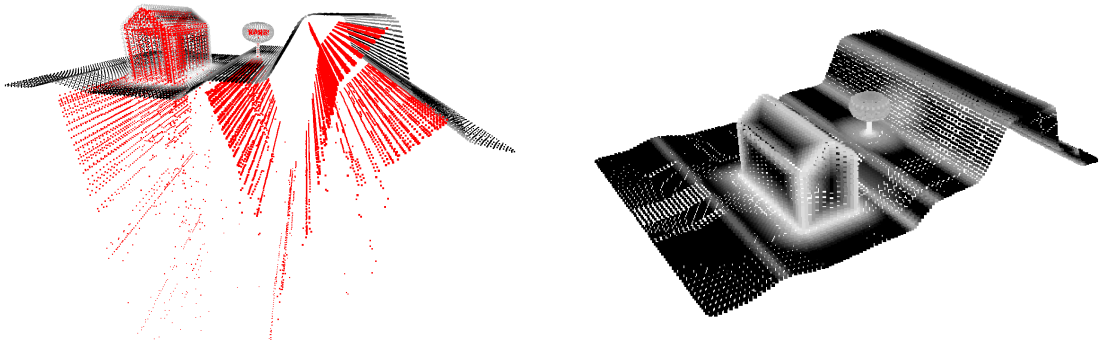
(a) Early illustration of the 2D MAT (consisting of edges and points) of two '1anthropomorphs' [Blum, 1967].

(b) The 3D MAT (right) of a surface model of a hand (left) consists of faces, edges and points [Amenta et al., 2001].

Figure 1.2: The Medial Axis Transform (MAT)

such as trees or glass). In other words: when the DSM features actual 3D geometries, problems arise. This is due to the planar triangulation that takes place during TIN decimation, which assumes that the input DSM is 2.5D. It clearly demonstrates the need for practically feasible DSM operators that treat the input DSM as truly 3D.

The Medial Axis Transform (MAT) is a shape descriptor that compactly represents both the geometry and the topology of shapes (see Figure 1.2). It is formally defined as the set of maximally inscribed (or *medial*) balls of a shape. The MAT was originally introduced by Blum [1967] and has found numerous applications in fields that deal with geometric structures, such as cosmology, geomatics, architecture, visual arts, computer vision and biology [Siddiqi and Pizer, 2008]. Consequently, it is a proven and established method for the analysis and recognition of shapes. Matuk [2006] researched the applicability of the MAT for feature-based terrain simplification. The core idea was to use the MAT to hierarchically decompose the terrain into features (such as summits, valleys and ridges) at different scales and to retract the corresponding skeleton branches to achieve selective feature generalisation. While the methods he implemented are based on a 2D approximation of the 3D MAT and seem to require continuous manual intervention, Matuk succesfully demonstrates that the MAT can be a valueable tool in the simplification of terrain models. His main result was that his method allows for very specific feature removal (generalisation) without significantly disturbing global surface characteristics such as the drainage network. This supports the idea that we can use the MAT to 1) decompose a DSM into a set of features at different scales, 2) conveniently traverse those features by using the topology of the MAT that connects them and 3) generalise a DSM by altering the configuration of MAT branches. Furthermore, if this could be done based on the *3D* MAT, we may have a good candidate framework for the analysis and processing of modern 3D DSMs. Unfortunately, up untill recently, the algorithms to approximate the 3D MAT were hampered by problems of robustness and scalability. But, due to a recently developed algorithm by Ma et al. [2012] and Jalba et al. [2012] (see Figure 1.3), a simple, robust and scalable implementation of the



(a) Synthetic dataset and point approximation of its interior 3D MAT (in red)

(b) Synthetic dataset colored by local features size, a measure of importance based on the MAT.

Figure 1.3: Approximating the 3D MAT using an own implementation of the shrinking ball algorithm by Ma et al. [2012]

3D MAT is possible now. Thus, a renewed exploration and an extension of the ideas that were originally coined by Matuk are justified.

1.2 Objectives

The goal of this research is to develop and prototype a robust and fully 3D method for the analysis and generalization of DSMs based on the 3D MAT. The hypotheses are that the 3D MAT gives the means to

1. enable a truly 3D analysis of modern DSMs, and
2. define DSM features using the geometrical (medial ball radii and geometry of MAT points) and topological (branch connectivity and correspondence between surface and MAT points) information encoded in the MAT.

The main objectives of this research are thus to investigate how the MAT can be used to perform:

1. DSM analysis, i.e. to compute importance measures on the DSM and to use these to recognize and identify application-specific features.
2. DSM generalisation, i.e. to simplify or to selectively remove application-specific features.

Chapter 2

Related work and background theory

2.1 State of the art in Digital Surface Models

We use geographic data models to represent what Goodchild [1992] calls *geographical reality*: empirically verifiable fact about the real world. These data models are limited representations of reality, constrained by the finite, discrete nature of computers. Through the use of such data models we are able to analyse and interpret geographical information on natural and artificial features in the terrain [Zhou and Zhu, 2013]. I define the Digital Surface Model (DSM), as an elevation model of the bare ground surface plus objects on top of it, thus as it is measured initially by Lidar for instance. In contrast, a Digital Terrain Model (DTM) represents, according to the definition I use here, just the bare earth surface, thus without any object such as vegetation and human-built structures on it [Li et al., 2010]. When considering DSMs, the Triangular Irregular Network (TIN) and especially the gridded Digital Elevation Model (DEM) have proven to be the most popular models over the last decades. These are classically seen as a discretization of an elevation *field*: an infinite set of tuples $\langle x, y, h \rangle$ that assign an elevation h to a location (x, y) [Goodchild, 1992]. Since a (x, y) location in the field only has one unique elevation assigned to it, the field can not be used to represent true three-dimensional topographic features such as caves, overfolds or vertical walls. For this reason a field is said to be single-valued or $2.5D$; only two of its spatial dimensions are independent. As argued by van Kreveld [1997], the choices for particular DSM and the algorithms to process it (e.g. viewshed analysis, contour extractions, and terrain simplification) are interwoven. The design of an efficient algorithm and the choice for a terrain model, and also the data *structure* that implements it, go hand in hand.

DSM
: an elevation model of the bare ground surface plus objects on top of it.

Gridded DEM
: a regular square grid of elevation values. Usually stored as a simple two-dimensional array in computer, where each element represents a pixel. The interpretation of a pixel varies, it can be interpreted as an area or as a point, and this is not always handled consistently [Fisher, 1997].

Furthermore, disadvantages include a rectangular data array regardless of the morphology of the surface and under- and over-sampling of different parts of a diverse study area [Pike et al., 2009]. However, because it is so trivial to use, it has become ubiquitous. DEMs are the prime object of study in the field of *geomorphometry*; the science of quantitative land-surface analysis, which has the goal of extracting parameters (descriptive measures of surface form, such as slope, aspect, or wetness index) and objects (discrete spatial features, such as watershed lines, alluvial fans, or drainage networks) from DEMs [Pike et al., 2009].

TIN

a finite set of points stored together with their elevation. As opposed to the DEM, the points need not to lie in a particular pattern and the density may vary. A planar triangulation is defined on the 2D projection of these points, and every triangle usually defines a linear interpolation between its three defining points.

2.1.1 3D Surface Models

As a result of advances in data acquisition techniques such as Lidar, point clouds (collections of points in 3D space) have become a primary source for the creation of detailed TINs and DEMs. But, a point cloud often contains information that can not be represented using the 2.5-dimensional field definition, upon which both the DEM and the traditional TIN are based. For example in the case of over-folded geological structures, buildings with balconies or simply buildings with vertical walls. When such a 3D point cloud is converted to a 2.5D gridded DEM or a TIN, this information is therefore lost and can no longer be used in any future analysis. However, it is possible to extend the definition of the TIN and the DEM to fully support 3D surfaces. In fact, a TIN is trivially extended to 3D by simply removing the condition that it needs to be projectable to the (x, y)-plane. This is commonly called a surface mesh, and it would not require any change in the data structure (but algorithms to process it are more complex). Methods exist to compute surface meshes automatically from aerial imagery [Haala and Kada, 2010]. A gridded DEM can also be extended to 3D by adding a third axis to the grid. This is called a voxel grid, and, if we naturally extend the definition of a pixel, each voxel (or *volumetric pixel*) is now a volumetric object. That is quite different from a polygonal mesh, which does not explicitly model volumetric elements; it only explicitly models the boundary (surface) of objects. Instead, the triangular counterpart of the Voxel grid is called a tetrahedralisation [**?**], which consists of tetrahedra decomposing the volume. Furthermore, the point cloud itself could also be considered a kind of three-dimensional DSM. Although, it lacks the connectivity of the earlier described models, and therefore does not really define a *surface* explicitly (note that it is much less trivial to reconstruct 3D surfaces as opposed to 2.5D surfaces, see for instance Heckbert and Garland [1997]). It is, however, arguably the simplest representation for a three-dimensional DSM imaginable, resulting in lower storage requirements when compared to for instance a polygonal mesh.

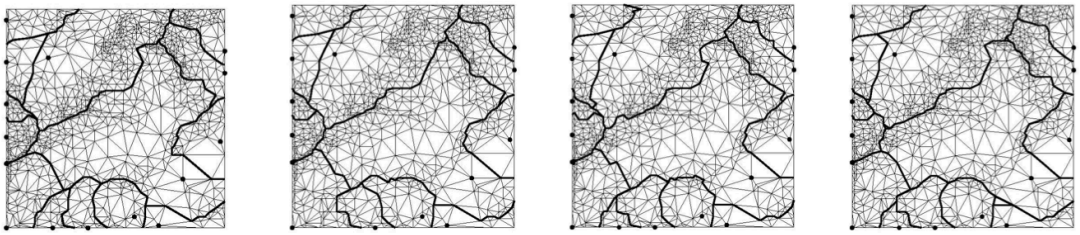


Figure 2.1: Comparison of watershed delineation algorithms based on Morse complexes by Magillo et al. [2009]. While all of them may look plausible, it is not easy to say which result is best.

2.1.2 Finding and reconstructing features in the terrain

This section lists a number of existing methods to extract features from a DSM or DTM. What is a feature? That depends stronly on the application of the method. Geometrically, a feature can be a point, a line, a surface, or a mixture of these.

The geometry or morphology of landscapes has been studied comprehensively since the nineteenth century [Wood, 1996]. It is commonly based on the computation and identification of simple geometric point and line features in the surface, such as local extrema, saddle points, ridges and ravines. These features are topologically related in a certain way, and this topology can be expressed in a graph which decomposes the terrain into different regions, with the simple features at the boundaries. More complex features, such as watersheds, can be identified through the analysis of such a graph (see for instance Magillo et al. [2009] and Figure 2.1). These methods are rooted in *Morse Theory* [Edelsbrunner et al., 2001], and can be implemented using a gridded DEM [Wood, 1996] or a TIN [Magillo et al., 2007]. It should noted that, while the results of these algorithm may appear to be plausible, different algorithms can give significantly different results (e.g. watershed delineation) for equal inputs [Magillo et al., 2009]. Related to this are methods that decompose the terrain based on contour lines [Kanade, 1994; Van Kreveld et al., 1997]. Based on how contour lines split and merge as a function of the elevation, the *contour tree* is constructed. Gauch and Pizer [1993] performs shape-based segmentation in a DEM using the *intensity axis of symmetry*, which is based on the MAT.

Recently, there has been a large interest for the automated extraction of building models from various types of input data. These methods typically deliver (textured) surface meshes for urban areas. Haala and Kada [2010] and Musialski et al. [2013] describe recent developments on this topic, which has applications in the entertainment industry (movies and games), digital mapping (Google earth, Apple Maps and Bing Maps), urban planning (e.g. line of sight analysis), and training and simulation (crisis management). Different methods differ in the type of input data that is used (Lidar and/or photographic imagery, either ground-based or aerial), the quality of input data, and the degree of automation. Clearly, the quality of the result depends on the quality of the input data. Furthermore, some methods also exploit the availability of building footprints, others sometimes derive these automatically. Haala and

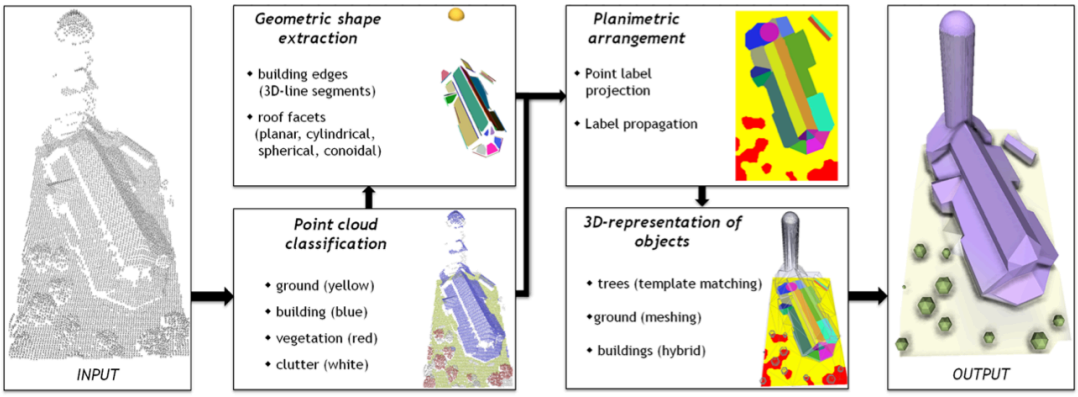


Figure 2.2: Urban reconstruction approach from Lafarge and Mallet [2012]

Kada [2010] identifies three main approaches for the reconstruction of buildings and their roof shapes. First, by assuming that buildings are simple shapes that can be represented using parametric shapes, whose parameters are estimated from the input data. Such an assumption may hold for most buildings in a dataset, but is also dependent on the country and what a common building looks like. Second, reconstruction can be performed based on a three step approach of segmentation, feature recognition and modelling. That means to segment for instance a point cloud into shapes, such as planes, spheres, and cylinders, detect topological relationships (boundaries) between these segments, and finally, to infer a geometric model from this information. Third, reconstruction by DSM simplification based on highly detailed surface meshes. The goal here is to simplify this DSM, by reducing the number of faces, while preserving the shape properties of man-made structures through the incorporation of constraints that are driven by planes, spheres, cylinders and cones that are detected in the input model. The edges of these shape primitives are preserved though out the simplification process, which is based on collapsing edges. While this classification is insightful, I believe they are in some cases overlapping. Notably, all approaches (rightly) presume certain conditions on how a building is supposed to look like. I think it is important to recognise the presence of this semantic knowledge in the modelling approach itself. Some approaches, such as the one by Lafarge and Mallet [2012] (see Figure 2.2), are also able to reconstruct parametrized vegetation.

As a concluding remark, It seems that a single universal definition of features in the DSM does not exist. One class of methods focuses on the recognition of natural landforms (geomorphometry), but it turns out to be quite difficult to objectively and precisely decompose a (generic) terrain into different landforms (see also Evans [2012]). On the other hand, we have the urban reconstruction algorithms that (at best) distinguish between bare-earth, (some types of) buildings and vegetation. These algorithms are specifically designed to extract building geometries, and impose semantic rules in the process of doing this. Finally, the results of both of these classes strongly depends on the quality and the form of the input.

2.1.3 Terrain simplification

DSMs are potentially large datasets, i.e.: they may represent a large region with high detail (densely sampled). For various reasons, but mostly to make them more manageable, it is wished to reduce the storage size of these datasets. The objective of terrain simplification is to minimise the the number of vertices in a DSM while staying as close as possible to the shape of the initial model. A TIN or a surface mesh is best suited for this form of simplification, since it has adaptive resolution: a simplified TIN has just enough vertices to model every phenomena it represents within a given tolerance. This means more vertices in regions of high variability and less vertices in (planar) areas of small variability, which is not possible with a gridded DEM. Tam and Heidrich [2002] call this *mesh-simplificaction*, as opposed to *feature* or *shape-simplification* which simplifies or generalizes a shape by selectively removing distinct features. Compare Figure 2.16 to Figure 2.19.

Garland and Heckbert [1995] conduct a comprehensive survey of numerous existing terrain, i.e.: elevation *fields*, simplification methods developed by researchers from various fields in science. They also propose their own fast algorithm to approximate height fields. It is based on a very simple local metric of judging the importance of a point, first used by Lee [1989] who proposed a comparable algorithm (he named it the drop heuristics algorithm), that in terms of approximation quality outperforms more complicated metrics. This metric is defined as the vertical (elevation) difference of an input point with the triangulation without that point at the same location. This metric calculated for all input points after which either the point with the biggest difference is added to an initial minimal triangulation that covers the data extent (this the *refinement* approach) or the point with the smallest difference is dropped from a complete triangulation of all points (the *decimation* approach). That step is repeated until a pre-set threshold of elevation difference is reached. As pointed out by Lee [1989], Garland and Heckbert [1995] and van Kreveld [1997] this results in a high quality approximation of the sampled field with only a limited set of vertices that is also a Delaunay triangulation.

Garland and Heckbert [1997] propose a truly three-dimensional simplification algorithm for polygonal surface meshes (see Heckbert and Garland [1997] for a survey of such methods), which as demonstrated in the same paper also works well for terrain models. The quadric edge contraction algorithm that they propose is based on a metric that is defined for each vertex as the sum of squared distances of the vertex to the planes that are spanned by the triangles incident to that vertex. For every edge in the triangulation the sum of this quadric metric for its incident vertices is calculated. And by minimising this sum, the location of a new vertex follows. The next step is to perform the edge contractions (replacing its two initial vertices with a single vertex that minimises summed quadric errors) for the edges for which the quadric error is the lowest. Evidently, every edge contraction also implies a local re-triangulation. This process is continued until only a preset number of triangles is remaining. The quality of resulting geometry in terms of approximation of the initial model is arguably better than the earlier described simplification algorithm, because quadric edge contraction is based on a fully three-dimensional metric and also optimises the location of new vertices, whereas drop-heuristics is just based on a one-dimensional metric and does not perform any optimisation often vertex location.

The amount of detail in the surface model that is required for a particular application may vary. Garland [1999] describe a number of discrete and continuous *multiresolution* models: hierarchical struc-

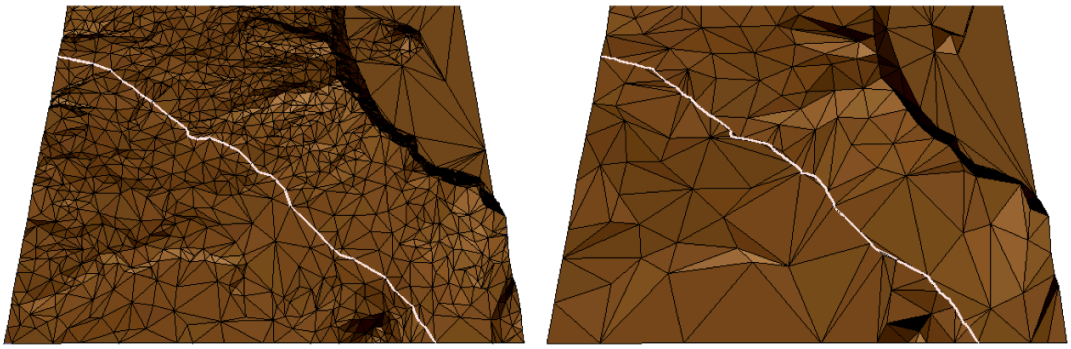


Figure 2.3: Terrain simplification with preservation of a user-defined break line [Kho and Garland, 2003].

tures that represent multiple levels of simplification for a given surface mesh. This makes it easy to obtain the preferred level of simplification for a given application, or to switch (continously) between different levels of detail. Of course, there is always a tradeoff between the quality and the computation time of a simplified mesh, and the various available methods give different compromises to this regard. Furthermore, some features which have a higher level or semantic meaning, such as ridge lines in a terrain model, can be difficult to preserve using an automated method. Therefore, Kho and Garland [2003] propose a user-guided simplification procedure that allows the user to indicate what features are important and need to be preserved during the simplification process (see Figure 2.3).

2.1.4 Dealing with huge datasets

Massive DSMs are now ubiquitous in geomatics. In fact, they are so large that they do not fit in a computers main memory, which makes it impractical to use many conventional algorithms. Therefore, some researchers have focused their efforts on the development of algorithms that are optimised to work efficiently with these huge datasets. As described in Isenburg et al. [2006a], some of the main strategies to do this are: divide-and-conquer, external memory, and streaming. Divide-and-conquer algorithms divide the input into parts, which are processed independently, and then put back together. Difficulties with these algorithms are: choosing the right way to cut the input, tedious programming may be require to communicate across cuts, and possibly poor-quality results near the cuts. See for instance the work of Constantin et al. [2010]; an approach simplify large meshes using quadric edge contraction. External memory algorithms on the other hand, use the hard disk to store temporary data structures (such as binary trees or octrees), and minimise the number of disk-accesses. Example are the work of Kreylos et al. [2008] for the visualisation of massive point clouds, Agarwal et al. [2006] for computing DEMs from point clouds and Arge et al. [2010] for cleaning massive multi beam echo sounding point clouds. Streaming works by performing a small number of sequential passes over the input data, and processes the data using a memory buffer whose size is a fraction of the stream length [Isenburg et al., 2006a]. While this limits the range of operators that can be applied—they must be local—memory requirements

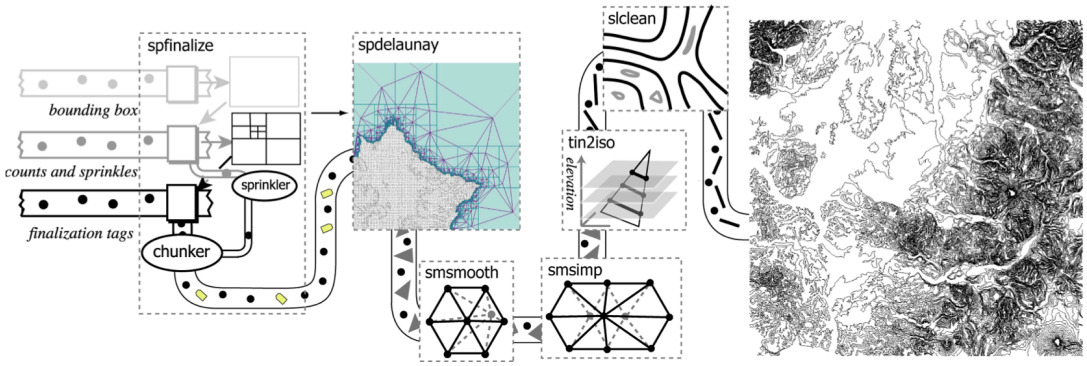


Figure 2.4: Streaming processing employed to extract smoothed contour lines from a massive point cloud [Isenburg et al., 2006c].

are extremely low. Another benefit is that, since the processed data is immediately outputted, the next processing step can already commence before the current one is finished (see Figure 2.4). As demonstrated by Isenburg et al. [2006c] and Isenburg et al. [2006b] streaming can be effectively applied to massive point clouds in order to e.g. to generate a Delaunay Triangulation, to perform smoothing, to extract contour lines, or to generate gridded DEMs. While the mentioned algorithms all take point clouds as input, similar algorithms exist for polygonal surface meshes [Isenburg and Lindstrom, 2005].

Aside from algorithms themselves, datastructures are also important for scalability, both in terms of storage as in terms of computational cost. Besides, often an algorithm is specifically designed for a particular datastructure, which means that the input data may need to be converted if it is not already in that format, which may be a costly operation on its own. As described in Section 2.1.1, a DSM can be stored using a number of datastructures. They are listed in Table 2.1 that compares the different datastructures in terms of required storage for equal resolution (defined along one axis, so 4 points/m would mean 16 points/m^2) and otherwise somewhat simplisitic assumptions (e.g. no compression). But it proves an important point. See Figure 2.5, it illustrates how the total required storage depends on the resolution. It is obvious that a 3D grid quickly requires massive amounts of storage, since it is actually a volumetric datastructure, whereas the other datastructures represent only a surface. Clearly, this needs to be considered when working with a (huge) DSM: two algorithm that both run in linear time are not equaly scalable if one requires voxel input and the other requires point input.

2.2 Shape analysis and matching

This section aims to provide a short overview of what is important for shape matching (determining how similar two shapes are), in the context of recognising shapes in a DSM based on the MAT, and is mostly based on the survey paper by Tangelder and Veltkamp [2008]. Shape analysis and matching is an important part of the proposed workflow of DSM analyisis (as described in Chapter 3).

Table 2.1: Comparison of storage requirements for DSM datastructures for a 1kmx1km tile. (with a height of 100m in case of the 3D grid.)

Type	Dimension	Storage per element	Number of elements	Total storage	Resolution
2D grid	2.5D	32 bit	16 M	61 MB	4 points/m
3D grid	3D	1 bit	6400 M	762 MB	4 points/m
3D point cloud	3D	96 bit	16 M	183 MB	4 points/m
2D TIN (star)	2.5D	288 bit	16 M	549 MB	4 points/m
3D TIN (star)	3D	288 bit	16 M	549 MB	4 points/m

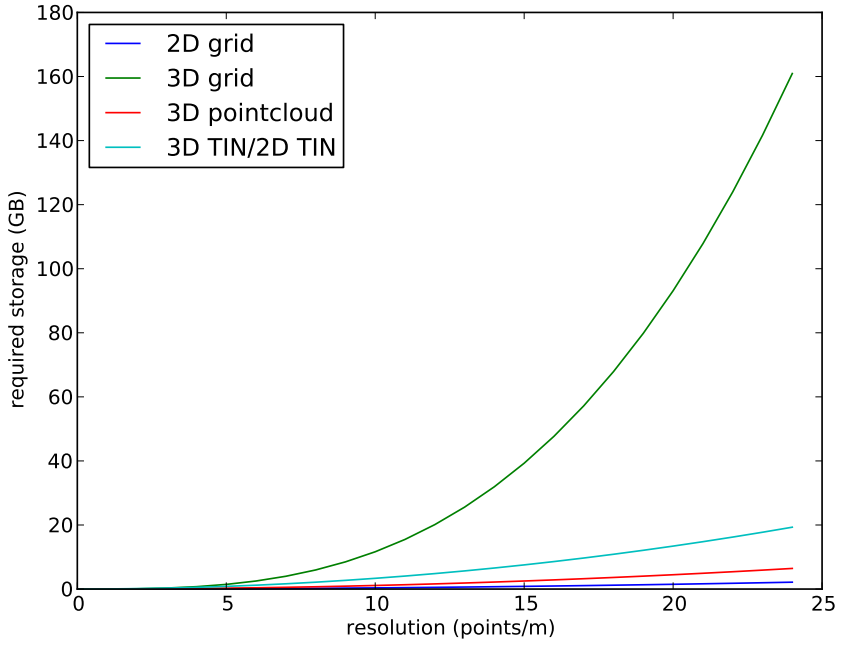


Figure 2.5: Storage requirements for a 1000x1000 m^2 surface tile as a function of a homogeneous resolution along each axis.

The process of shape matching requires a shape descriptor: a compact overall description of shape. A variety of shape descriptors exist, they mainly differ in what they measure (for example point-to-point distances, volume, curvature or topology), how they represent it (for example as an image, a set of vectors or a graph), and whether they measure local or global properties of shape. A *dissimilarity measure* can be defined that quantifies the similarity between shapes by measuring some notion of 'distance' between a pair of shape descriptors. Evidently, the smaller the value of the disimilarity measure, the more identical two shapes are. The effectiveness of a dissimilarity measure depends on the type of shape descriptor and may be strongly application dependent. In general, a number of issues can be identified here:

Discriminative power
how well does a shape descriptor capture discriminative properties of an object's shape.

Partial matching
graph-based shape descriptors are able to find partial matches: a part of a shape that is similar to a part of another shape. Figure 2.6b illustrates how a shape can be hierarchically decomposed into different parts based on the MAT. This is also relevant for the recognition of (natural) feautures in a DSM.

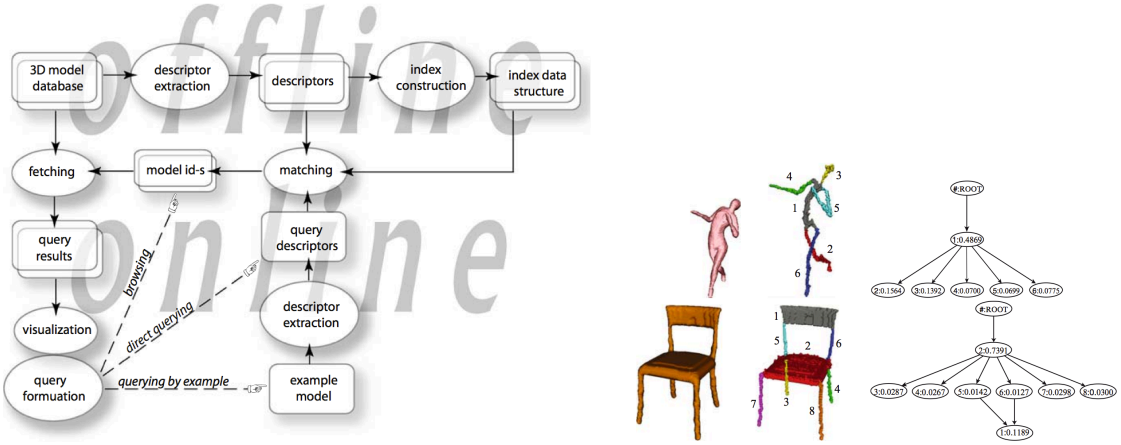
Robustness and sensitivity
for many application it is desirable for a shape descriptor to be insensitive to noise and relatively small features on a shape. However, when a descriptor is too insensitve to noise, it is probably also not very sensitive, which in turns negetively impacts the discriminative power.

Pose normalization
In general an object may have an arbitrary scale and rotation, which may need to be normalized in order to compute a usable shape descriptor. However, for graph-based shape desciptors this is not so much of an issue. Furthermore, when we consider georeferenced objects objects such as houses of certain dimensions, a scale-dependent shape descriptor may actually be desirable.

Furthermore, Tangelder and Veltkamp [2008] describe a conceptual framework for shape retrieval (see Figure 2.6a) that allows to quickly match a query shape to a database of pre-computed shape descriptors. It uses a special index of shape desciptors and assumes that dissmimilarity measures can be computed relatively efficiently.

2.3 The Medial Axis Transform

What is a shape? Among the several plausible definitions, Chang [2008] lists this one: "A shape is all geometric information of an object invariant to Euclidean transforms (e.g., translation and rotation)". The MAT is in essence a *representation* of shape. Different representations of shape exist, and their usefulness depends on how and for what purpose they are used. In GIS, shapes are commonly represented using a set of points, lines and surfaces in a way that explicitly delineates the boundary of a shape. This seems like a very natural and intuitive way to deal with shapes, and it is certainly very efficient for the



(a) Conceptual framework for shape retrieval [Tangelder and Veltkamp, 2008]

(b) Hierarchical decomposition of a shape by means of the MAT [Siddiqi and Pizer, 2008]

Figure 2.6: Shape analysis and matching.

purposes of storage and visualisation. The Medial Axis Transform (MAT) [Blum, 1967], on the other hand, makes the *topology* of a shape explicit as a hierarchical structure. This is particularly useful for applications such as shape analysis and matching that benefit from the ability to quantify and distinguish different parts of a shape.

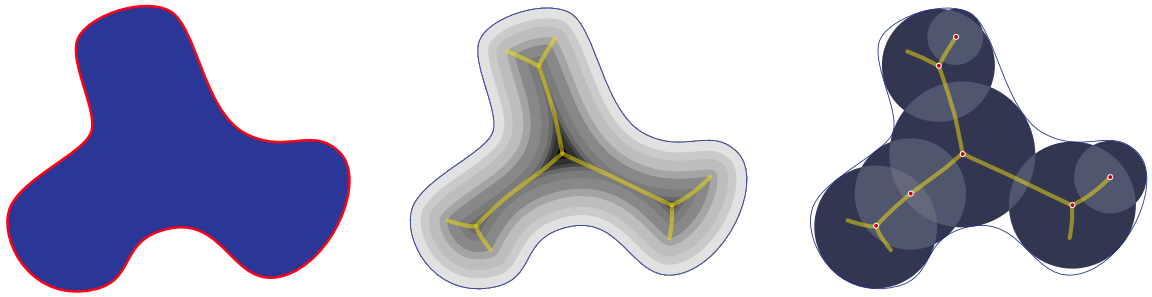
In the following sections, I first define the MAT and list its properties. Second, I present an overview of different methods to approximate it. And finally, I discuss methods to simplify or *prune* the MAT.

2.3.1 Defining the Medial Axis Transform

As already indicated by Blum [1967, 1973], who originally introduced the concept, a number of alternative ways exist to define the concept that he originally named the MAT, but is sometimes also referred to as *symmetry axis* or simply *skeleton*. These definitions usually start with the boundary of a shape X in $\mathbb{R}^d$. For this research I will generally assume that $d = 3$, but in order to effectively explain some of the basic concepts that underly the MAT, I will also discuss cases for which $d = 2$.

Grassfire propagation

In a very intuitive definition of the MAT a grassfire analogy is used. Consider the boundary of a shape X (marked red Figure 2.7a), and imagine that all the points on this boundary are simultaneously set on fire at time $t = 0$. The fire spreads evenly to all directions at constant speed. Now, the MAT is defined as the set of points where the fire front meets itself (i.e. where sharp edges start to appear). This concept is illustrated in Figure 2.7b, where the fire fronts of different times $t > 0$ are coloured with different shades of grey.



(a) A 2D shape

(b) Inner MAT according to grassfire propagation.

(c) Inner MAT according to locus of centres of medial balls.

Figure 2.7: Different definitions for the MAT of a shape.

Medial balls

A different definition, that nonetheless delivers the same MAT, is based on so-called medial balls. These are the balls that fit inside X and do not contain any other ball that would fit inside X. The MAT can now be defined as the set of points that are the centres of the medial balls (see Figure 2.7c). Alternatively, since a medial ball relates a point on the MAT to at least two closest boundary points, the MAT is sometimes defined as the set of points that have at least two to closest boundary points.

Furthermore, the points on the MAT are enriched with the radius of the corresponding medial ball radius (or, equivalently, in the grassfire definition with the timestamp at which the point emerged). Formally I use $\mathcal{M}[X]$ to denote the MAT of an object X.

For concave shapes, the MAT is made up out of two parts: an inner part, which completely resides inside X (this part is depicted in Figure 2.7), and an outer part, which emerges from the concavities in X and resides completely outside X. For 2D shapes, the MAT is a 1D structure (embedded in 2D space): it is consists of points and lines only. However, for 3D shapes, the MAT is a 2D structure (embedded in 3D space): it consists of points, lines and surfaces. These surfaces are sometimes called *sheets*. Figure 1.2b gives an example of the 3D MAT.

2.3.2 Properties and characteristics of the Medial Axis Transform

The MAT has a number of valuable properties, e.g. for shape analysis, these are listed here:

Complete

The MAT completely describes the shape of object. As a result, we can not only compute the MAT from a boundary representation of X, but we can also reconstruct that boundary using solely the information that is present in the MAT.

Topology preserving

This means that the object boundary can be continuously deformed into the Medial Axis without change in genus.

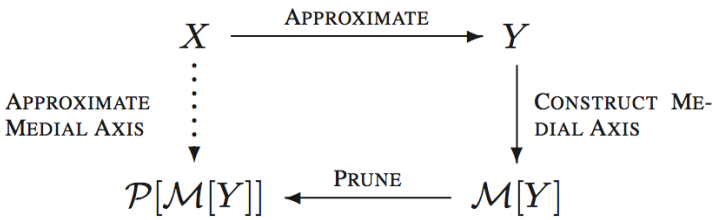


Figure 2.8: Approximating the MAT [Biasotti et al., 2008]

Compact
The MAT is generally of dimensionality $d - 1$, hence one dimension lower than X itself, and the space in which it is embedded. In general, structure of a lower dimensionality are easier to analyse and process.

Hierarchy
The structure of the MAT enables an hierarchical traversal of the different parts (each part corresponds to a MAT-sheet) that define an object.

Medial
The MAT is centred exactly in the middle of a shape.

Sensitivity
Small perturbations in the object boundary may cause large perturbations in the Medial Axis of that object. This means the MAT is extremely sensitive to noise in theory.

2.3.3 Computing the Medial Axis Transform

Computation of the MAT has been a ongoing research topic since the MAT was originally introduced by Blum. Biasotti et al. [2008] and Attali et al. [2009] argue that, while methods for the exact computation of the MAT do exist, they are not feasible in practice. Exact computation is only possible for a class of shapes termed *semi-algebraic sets*, i.e. solutions of finite systems of algebraic equations and inequalities. For these shapes, the MAT itself is also semi-algebraic, it is in fact composed of parts of the bisectors of the algebraic functions that define the shape. Computing these bisectors proves to be difficult even for shapes bounded by simple curves in a plane.

However, we can also *approximate* the MAT. This is usually much less expensive in computational terms and gives satisfactory results for arbitrary shapes. The trick is to find an approximation Y of the object X, for which the MAT can be computed exactly and efficiently. Miklos et al. [2010], for instance, state explicitly that they choose to approximate X with the union of a finite number of balls. Of course this is not an arbitrary choice, because this particular subset of shapes, named *polyballs* by Attali and Montanvert [1997], is very closely related to the definition of the MAT: every ball centre defines a point on the MAT. Thus, a polyball approximation Y allows us to compute $\mathcal{M}[Y]$ exactly and in a practically feasible manner. Finally, by simplifying or *pruning* $\mathcal{M}[Y]$, we obtain $\mathcal{P}[\mathcal{M}[Y]]$, a subset of $\mathcal{M}[Y]$,

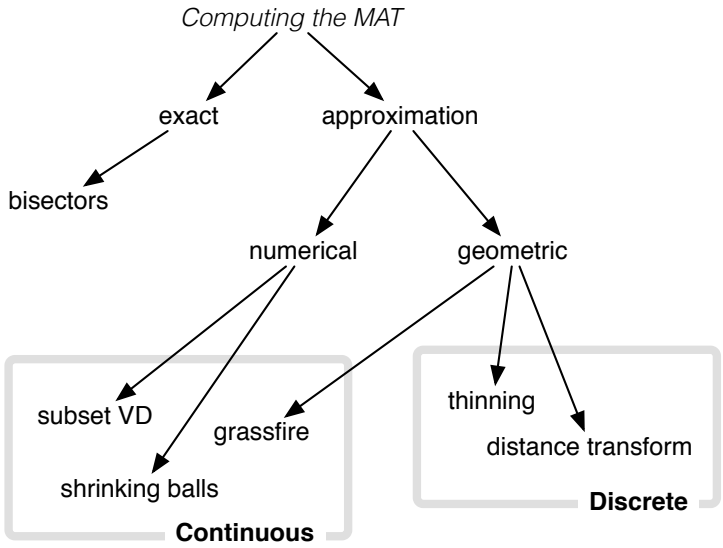


Figure 2.9: Classification of methods to compute the MAT

that aims be an adequate approximation of $\mathcal{M}[X]$. This step is required because Y is, after all, an approximation of X and may contain noise or other inaccuracies which may significantly disturb $\mathcal{M}[Y]$. The complete process of approximating $\mathcal{M}[X]$ is summarised in Figure 2.8.

Biasotti et al. [2008] make a classification of approximation algorithms based on two different criteria, see Figure 2.9. First, a MAT approximation algorithms is either *numerical* or *geometric*. The output of numerical algorithms is always near $\mathcal{M}[X]$ and possibly exactly $\mathcal{M}[X]$ when the sample Y of X is sufficiently dense and accurate. In contrast, a geometric algorithm is a new descriptor that is defined in a similar but different way, when compared to the MAT itself. Thus even with perfect sampling this class of methods will not converge to $\mathcal{M}[X]$ exactly. Second, MAT approximation algorithms are either *continuous*, if they manipulate and output points with real (continuous) coordinates, or *discrete* if the result is obtained through the application of regular discrete grids. While I believe this classification is useful, complete and insightful, I choose to focus on algorithms that are practically feasible and divide them in different groups based on their main methodology. These are

1. algorithms based on the continuous Voronoi diagram,

2. algorithms based on the shrinking balls paradigm, and

3. algorithms based on the discrete distance transform.

For the sake of completeness, I will first briefly discuss some algorithms that have also been implemented, but are mostly interesting from a theoretical rather than a practical point of view.

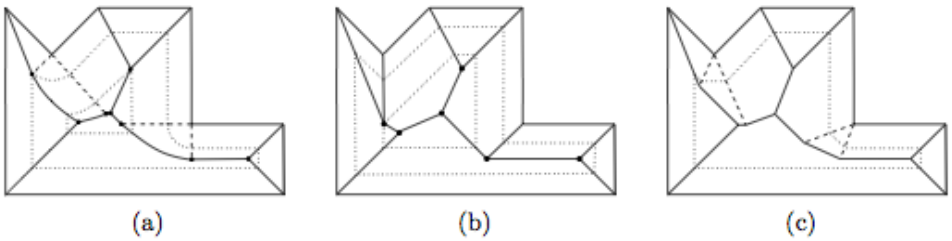


Figure 2.10: Medial Axis (a), straight skeleton (b) and linear axis (c) for a 2D polygon [Tănase and Veltkamp, 2004]

Theoretical methods

The *straight skeleton* is an implementation of the grassfire propagation approach. It was introduced by Aichholzer et al. [1996] for 2D input and later extended for 3D input by Barequet et al. [2008]. The straight skeleton is computed by letting a wavefront of straight polygon edges propagate inwards. During this process the wavefront edges remain parallel to the boundary of the polygon. Events are generated when wavefront edges disappear. Possible events occur where the bisectors of polygon vertices intersect. By detecting these events in the right sequence the vertices and the connectivity of the straight skeleton is computed. As observable from Figure 2.10, the straight skeleton is quite similar to the MAT, except that it only has straight line segments and that it has branches that connect to concave vertices on the polygon boundary. The *linear axis* [Tănase and Veltkamp, 2004] is a variant of the straight skeleton that prevents the creation of counterintuitive edges, and thus is a better approximation of the MAT. Barequet et al. [2008] report on implementations of the straight skeleton for the 3D case of so-called voxel polyhedra (polyhedra with integer coordinates and faces parallel to two of the coordinate axis) and orthogonal polyhedra (that just have axis-parallel faces). For general polyhedra, they point out that the straight axis is ambiguous. Furthermore, I would expect problems with computational robustness for real world data. Also note that it is not possible to reconstruct using medial balls the original object from the straight skeleton, which is important for this research.

Continuous Voronoi methods

Given a sufficiently dense point sample S of the boundary of a shape X, it can be observed that a subset of the Voronoi Diagram (VD) of S approximates $\mathcal{M}[X]$. Brandt and Algazi [1992] gave a proof for the two-dimensional case. Observe from Figure 2.11 that those VD edges that do not intersect the boundary of X contribute to $\mathcal{M}[X]$. Furthermore, following the duality between the VD and the Delaunay Triangulation (DT), if we take the dual of the remaining (boundary-intersecting) VD edges, we end up with an approximation of the boundary that is sampled by S. Attali and Montanvert [1997], Amenta et al. [1998a] and Gold and Snoeyink [2001] have used this duality to design algorithms that compute

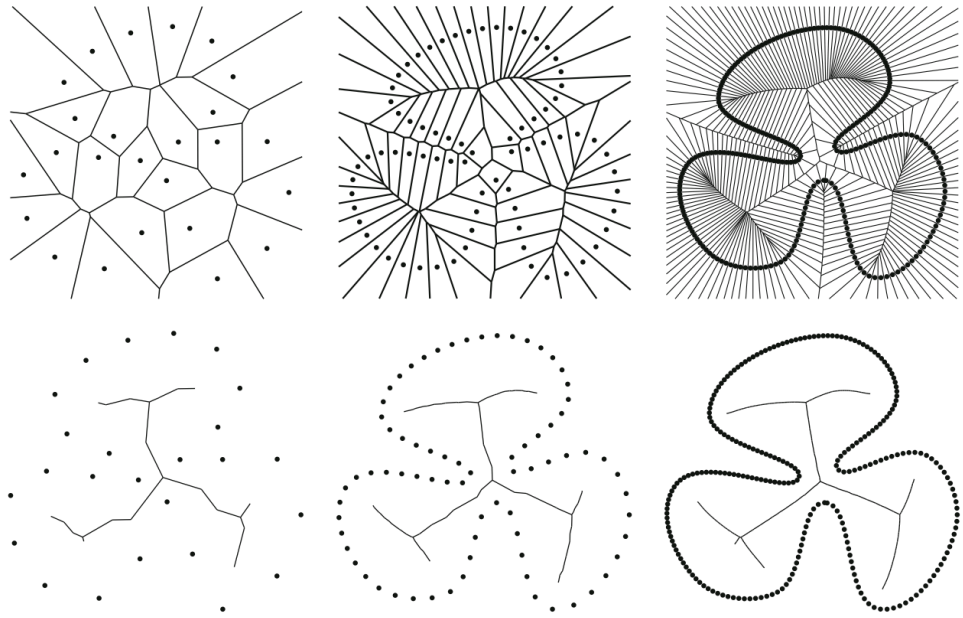


Figure 2.11: The VD can be used to approximate the MAT: the quality of the approximation (bottom row) increases with an increasing density of the boundary sample (top row). Sampling density increases from left to right. [Attali and Montanvert, 1996]

both this approximation of the boundary of X, as an approximation of $\mathcal{M}[X]$. The approach by Gold and Snoeyink [2001] uses a simple local test to construct these. However, these results are not easily extended to the three-dimensional case. As explained by Amenta and Choi [2008], the main difficulty is the presence of *slivers* in the 3D VD. These are almost flat Delaunay tetrahedra that appear when four or more sample points on the boundary of X are (almost) co-circular, which is quite common in practice. Unlike the two-dimensional case, the Voronoi vertices corresponding to these slivers lie far from $\mathcal{M}[X]$. Note that these slivers are unrelated to the presence of noise in the boundary sample S, in fact an arbitrarily dense sampling will result in Voronoi vertices that are arbitrarily far from $\mathcal{M}[X]$. As a result, the part of the VD that corresponds to these slivers needs to be filtered out before the MAT of a 3D shape can be approximated using the VD.

I would argue that three main approaches exist to approximate the 3D MAT based on the VD. One is by iteratively peeling away Voronoi cells starting from the boundary of an object. In the method of Attali and Montanvert [1997], cells are only removed if that does not change the topology of the MAT. Furthermore, they implement a threshold for filtering slivers. A second approach is based exclusively on filtering the elements of the VD. To achieve this, Dey and Zhao [2004] use two different local criteria that are notably independent of scale and sampling density. This results in an approximated MAT that is quite insensitive to noise. Unfortunately the outputted MAT may have holes, which is topologically incorrect. The third approach was presented by Amenta et al. [2001]: the power shape. Instead of using

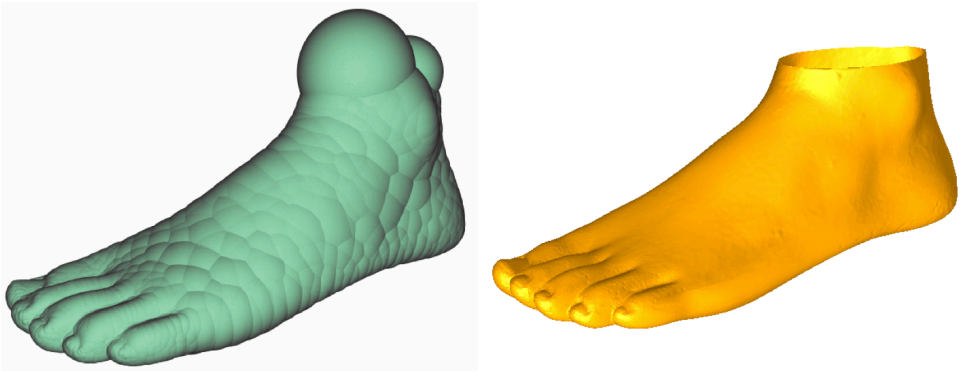


Figure 2.12: Reconstruction of the boundary of an object from just its MAT. Left: using a union of a finite number of medial balls. Right: using the power crust method. Figure from [Amenta et al., 2001].

the VD directly to construct the MAT, they use it only to find *poles*: the farther vertices of the Voronoi cell of a sample point on the boundary of X. A weighted VD, i.e. the *power diagram*, of these poles (weighted with the radius defined by their Voronoi balls) is constructed. The dual of this diagram, the *regular triangulation*, approximates the MAT of X. Notice that the power diagram can be used to reconstruct the surface of X, from the MAT, which is an interesting achievement on its own and I think it could also be used to reconstruct an object surface from a MAT that is created using another method, although it would require medial balls that are *outside* X as well. It certainly results in a much more natural looking surface than simply taking the union of medial balls (See Figure 2.13). A drawback from the power shape method is that it requires twice the calculation of a Voronoi-like diagram, whereas other Voronoi-based methods only require one such calculation. Furthermore, the output includes tetrahedra, which may be problematic for further processing. Tam and Heidrich [2003] therefore choose to extend and improve an earlier variant of the power shape algorithm, which outputs many duplicate geometries but does not exhibit the tetrahedron problem [Amenta and Kolluri, 2001]. Miklos et al. [2010] further improved the robustness of the algorithm and simplified parts of it by assuming the input to be a surface mesh rather than a point cloud.

Following is a list of properties that are generally true for MAT algorithms based on the continuous VD.

- They may exhibit numerical degeneracies. Jalba et al. [2012] are eager to point this out for the method of Miklos et al. [2010], supposedly one of the best continuous Voronoi-based MAT approximation algorithms (implemented in CGAL[1], but without costly robust predicates).

- Another recurring issue with this class of algorithms is the inability capture sharp edges that may be explicitly present, but insufficiently sampled, in the input model.

[1]http://www.cgal.org/

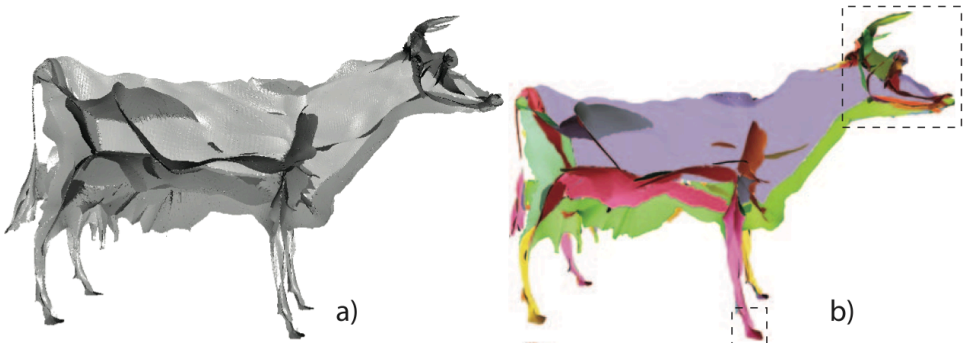
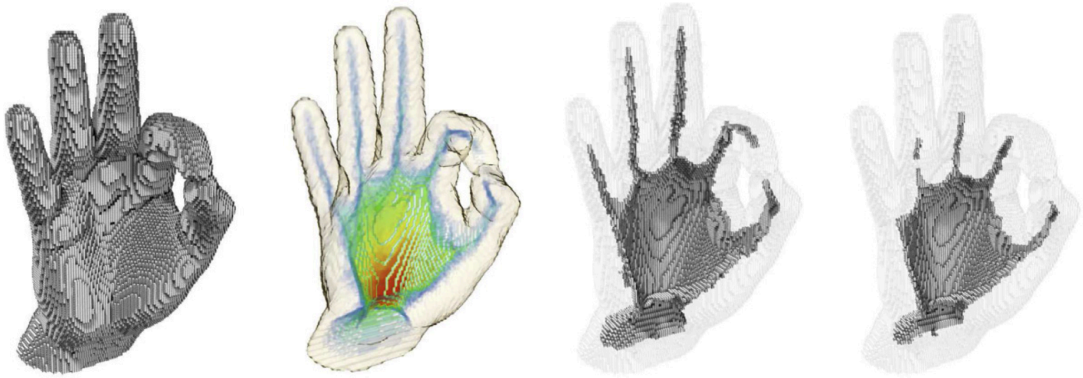


Figure 2.13: Two different approaches to reconstruct a surface MAT in the method of Jalba et al. [2012]. Left: Delaunay reconstruction, right: reconstruction based on clustering of sheets (rendered in different colours).

- The computation of the three-dimensional VD, which is the most expensive operation in this class of algorithms, has a worst case running time complexity of $O(n^2)$. However, Amenta et al. [2001] empirically demonstrate that for real-world input, the running time scales approximately linear. This is also verified by Ma et al. [2012].
- They are the most complex to implement, when compared to the other two classes of MAT approximation algorithms.

Shrinking ball methods

Ma et al. [2012] introduced the shrinking ball algorithm. Like the Voronoi-based methods, it outputs continuous coordinates, but it is based on a much simpler idea. It takes an oriented (i.e. including normals) point cloud as input, and outputs a set of (disconnected) points that represent $\mathcal{M}[Y]$. If normals are not available, they could be estimated using, for instance, the work of Hoppe et al. [1994]. The shrinking ball algorithm is based on the assumption that the centre of the medial ball corresponding to a sample point s must be positioned somewhere along line through the normal of s. Starting with a large ball B, B is iteratively shrunk until it neither touches nor contains any other points than s and only one more sample point. During this process a KD-tree is used as an efficient spatial index. The simplicity of the algorithm makes it very efficient and scalable. Furthermore, Ma et al. [2012] mention that it deals surprisingly well with sharp features in the input model. Compared to the power shape method of Amenta et al. [2001], it is between 2 and 15 times faster. An additional speed up of 5 to 10 times is gained when using a GPU implementation, and [Jalba et al., 2012] have improved the algorithm further, gaining roughly another order of magnitude. Additionally, they overcome the main limitation of Ma et al. [2012] (disconnected MAT), by applying one of two possible surface reconstruction techniques. The first one (which requires the input to be a mesh) projects MAT points on the corresponding triangle in the input mesh, and then creates a Delaunay Triangulation of these points, which then implies how



(a) Input model (b) Volume rendering of the distance transform. (c) λ = 11 voxels (d) λ = 15 voxels

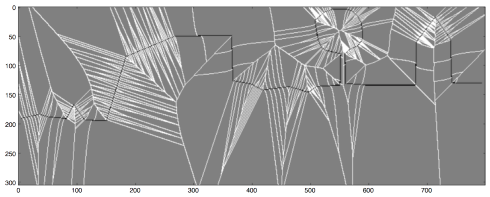
Figure 2.14: MAT extraction using the discrete λ-MAT method of Chaussard et al. [2011]. For Fig. 2.14b: Blue-green-yellow-red in order of increasing distance.

to connect the MAT points. This algorithm runs in $O(n)$ time, where n equals the number of triangles in the input mesh, and the authors note that it could be implemented using an out-of-core algorithm (this promises good scalability for large input models). The second method does not poses these favourable computational properties. It is based on clustering the MAT points into sheets, and uses a ball-pivoting algorithm to reconstruct the surfaces of the individual sheets. Concluding, the general properties of the shrinking balls method are:

- simple implementation,
- good scalability,
- robust,
- deals well with sharp edges in input model,
- requires no heavy data structures during computation (only a kd-tree of input points, which has low storage requirements when compared to Voronoi methods [Ma et al., 2012]), and
- requires in advance the computation of point normals.

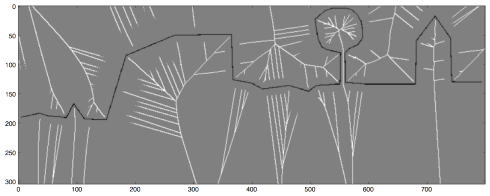
Taking a point sample of an object

The continuous Voronoi methods and the shrinking ball methods that have been described so far both take a point sample as input. What should such a point sample look like? Too few points might not adequately capture all features of shape, and too many points will take too long to process. The ϵ-sample, proposed by Amenta et al. [1998b], attempts to define a minimum sample of object that takes



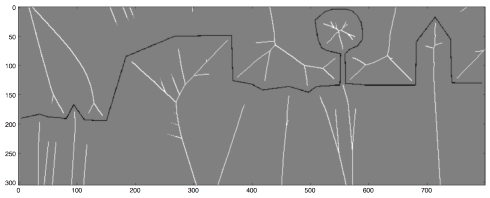
(a) MAT without pruning ($\lambda = 0$)

(b) Reconstructed surface without pruning.



(c) MAT with medium pruning ($\lambda = 6$)

(d) Reconstructed surface with medium pruning. Despite the simplified MAT, the corresponding boundary surface remains almost unchanged.

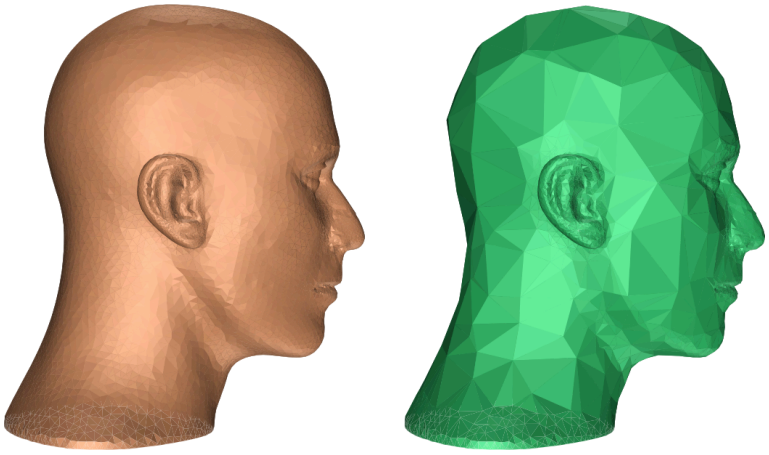


(e) MAT with strong pruning ($\lambda = 10$).

(f) Reconstructed surface with strong pruning. Notice how the tree is separated from the the ground and that edges have become rounder.

Figure 2.15: The effect of different levels of pruning on the 2D MAT. Own implementation based on a variant of the λ-MAT described by Hesselink et al. [2005].

into account the differences in the level of detail in different parts of the same shape. It is based on the *local feature size* $f(p)$, which is defined as the minimum distance from a point p on the boundary of an object X to $\mathcal{M}[X]$. Now, a point sample P of the boundary of X is an ϵ-sample if distance from any point on the boundary of X to the nearest point $p \in P$ is at most $\epsilon f(p)$. Many MAT approximation methods [Amenta et al., 2001; Dey and Zhao, 2004; Ma et al., 2012] assume the input to be an ϵ-sample, because it ensures that fine details are sufficiently sampled and it is often used to mathematically prove that some approximation method converges to the true MAT when $\epsilon \to 0$. An ϵ-sample can be obtained from an oversampled point sample of an object by using the work of Dey et al. [2001] or, and with more efficiency, the work of Ma et al. [2012]. In fact, this is an effective method to perform feature aware point decimation based on the MAT. And it has the additional advantage that the ϵ-parameter is scale-independent, in practice a value of $\epsilon = 0.4$ works well for datasets obtained by laser scanning [Dey et al., 2001]. See Figure 2.16 for an example.



(a) Surface reconstructed from over-sampled point sample (12772 points)

(b) Surface reconstructed from ϵ-sample with $\epsilon = 0.4$ (4272 points)

Figure 2.16: Decimation of a point sample using local feature size preserves points were needed. [Dey et al., 2001]

Discrete distance transform methods

Roughly two types of methods exist to approximate the MAT in a discrete manner: based on thinning or based on the distance transform. Thinning methods take a binary (voxel) image as input, and iteratively peel away layers of voxel at exterior of the object. A 3D example is described by Borgefors et al. [2008]. A prime feature of thinning is the preservation of topology, however the shape of the resulting skeleton often depends on the order in which pixels or voxels are thinned. Another large group of discrete MAT-approximation methods is based on the distance transform: an image where each element is labeled with the shortest distance to the boundary of an object X. Different measures for distance can be used and will result in different skeletons. From the distance transform the MAT can be intuitively recognised as a set of 'ridges': lines that indicate local directional maxima. Pixels or voxels that contribute to the MAT-approximation can be identified in a number of ways. Foskey et al. [2003] take the gradient field, a derivative of the distance transform, where every element consists of a unit vector in the direction of its closest boundary point. For every pair of adjacent vectors, the angle between them is computed and if this angle is larger than a threshold θ a facet between them is added to their MAT-approximation; the θ-SMA. Foskey et al. [2003] present in fact two variants of this algorithm: one based on a regular grid, and one based on a octree subdivision. The latter is more memory efficient and more time efficient, but is harder to implement and harder to parallelise. Sud et al. [2007] use the same angle criterium, but on MAT sheets rather than voxels. They compute the VD through the distance transform (with respect to a polyhedral input) and perform topology-preserving thinning on a graph that is derived from the VD and capture the connectivity between the MAT elements. A different criteria is chosen by Hesselink et al. [2005] and Chaussard et al. [2011]; they set a threshold on the distance between the two boundary points corresponding to two adjacent voxels in the distance transform. This is a discrete version of the

continuous λ-MA which was originally introduced by Chazal and Lieutier [2005]. And unlike their continuous counterpart, the discrete implementations do not offer any guarantee on the preservation of topology, as observable from Figure 2.15.

Following are some concluding remarks about the pros and cons of these discrete distance transform methods.

- A general criticism on discrete methods is that their output is not rotation invariant: i.e., spurious branches in the skeleton may appear when the object is not axis-aligned [Borgefors et al., 2008] (Figure 2.15a illustrates this).
- Sobiecki et al. [2013] state that voxel-based skeletization methods in general may also suffer from sub-optimal centeredness, i.e.; the axis is not precisely medial.
- However, their main concern is scalability; both in terms of storage as in terms of speed. While voxel-based methods are relatively fast for small inputs, with inputs that have a resolution of 1000^3 voxels or more, the computational cost are significantly slower since these algorithms usually scale linearly with the number of input voxels.
- On the other hand, these methods are generally (when using regular discrete grids) very easy to parallelise.

2.3.4 Pruning the Medial Axis Transform

Virtually all approximation algorithms apply some form of pruning (see Figure 2.8). This is necessary because of (small) differences in Y with respect to X, and the intrinsic instability of the MAT itself (e.g. the second image in Figure 2.17). But pruning can also be used to simplify the MAT past its natural shape, for instance to identify the most prominent parts of a shape (see the two rightmost images of Figure 2.17).

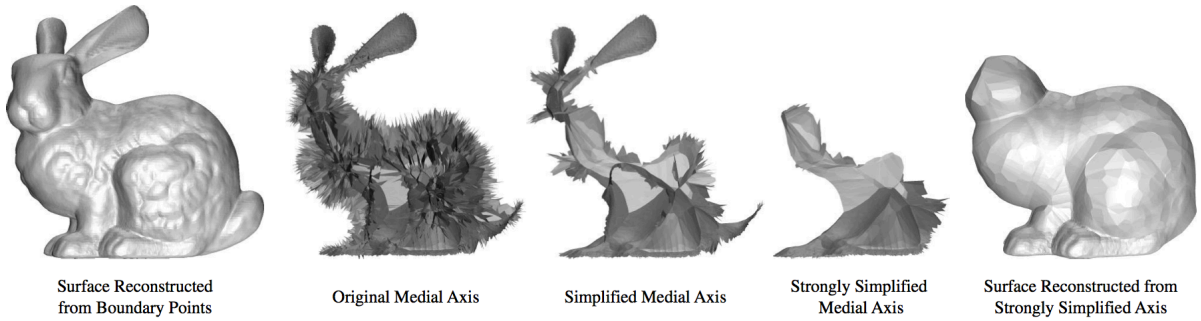


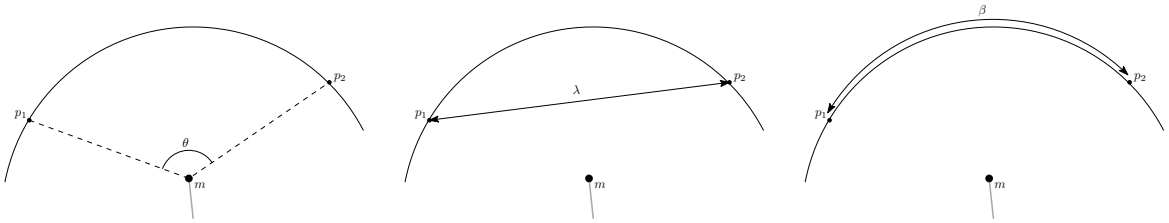
Figure 2.17: Simplifying the MAT [Tam and Heidrich, 2003].

In literature, I identify the following goals for pruning:

Table 2.2: Overview of algorithms to construct the MAT

Reference	Method	Input	Output	Worst-case time complexity	Topology	Robustness	Pruning	Other
[Attali and Montanvert, 1997]	VD peeling	points	mesh	$O(n^2)$	yes	sensitive to numerical instabilities	yes, limited	difficulty with sharp corners
[Dey and Zhao, 2004]	VD filtering	points	mesh	$O(n^2)$	no guarantee	sensitive to numerical instabilities	yes, limited	difficulty with sharp corners, filtering independent of density and scale
[Amenta et al., 2001]	power diagram	points	mesh	$O(n^2)$	yes	sensitive to numerical instabilities	yes, limited	sliver tetrahedra in output, difficulty with sharp corners
[Tam and Heidrich, 2003]	power diagram	points	mesh	$O(n^2)$	yes	sensitive to numerical instabilities	yes, sheet-based	based on Amenta and Kolluri [2001], shape-simplification
Scale Axis [Miklos et al., 2010]	power diagram	mesh	mesh	$O(n^2)$	depends	sensitive to numerical instabilities	yes, using scaling	based on Tam and Heidrich [2003], difficulty with sharp corners, requires watertight surface mesh
[Ma et al., 2012]	shrinking ball	points	points	$O(n^2)$	no	good	yes, based on local feature size	handles sharp features well, parallelizable, protruding balls in case of holes, requires normals
[Jalba et al., 2012]	shrinking ball	mesh or points	mesh or points	$O(n^2)$	yes	good	yes, based on shortest geodesic distance	based on Ma et al. [2012], requires normals, easy to parallellize, compares favourably to [Miklos et al., 2010], could be done out-of-core (up to reconstruction of surface skeleton)
θ-SMA [Foskey et al., 2003]	DT	mesh	voxels	$\Theta(n \cdot v)$*	no	good	yes	parallelizable, variant uses octree subdivision (less parallelizable)
θ-HMA [Sud et al., 2007]	VD based on DT	mesh	mesh	complicated	yes	sensitive to degenerate configurations	yes, sheet-based	implementation seems non-trivial, compares to Foskey et al. [2003]
IMA [Hesselink et al., 2005]	DT	voxels	voxels	O(n)	depends	not rotation invariant, noise due to discretization of input	yes	parallelizable
DLLMA [Chaussard et al., 2011]	DT	voxels	voxels	O(n)	depends	not rotation invariant, noise due to discretization of input	yes	parallelizable, improves Hesselink et al. [2005]

* where n equals the number of primitives in the input mesh, and v the number of voxels in the output



(a) θ-MAT (b) λ-MAT. (c) Boundary potential.

Figure 2.18: Common pruning measures from literature. They all define a metric based on the two boundary points p_1 and p_2 that correspond to the same medial axis point m

1. to remove unwished spurious branches (caused by noise or approximation errors) in the MAT,
2. to quantify the importance of different parts in the MAT, and
3. to remove parts of the MAT that correspond to some salient feature in the input shape.

Often a pruning method is applied to achieve a combination of these goals, and in some cases it is tightly integrated with a particular MAT approximation algorithm (especially the discrete methods). Shaked and Bruckstein [1998] perform a study of pruning methods in 2D. In the following paragraphs, I describe and exemplify three groups of prominent 3D pruning methods.

Continuous pruning methods

are the most common. They usually define an importance measure for every point on the MAT based on some geometric property. Next, the MAT points are thresholded with this measure. The resulting (pruned) MAT $\mathcal{P}[\mathcal{M}[Y]]$ is usually a subset of $\mathcal{M}[Y]$. Some methods preserve topology, others do not or only up to a certain level. The main challenge is often selecting the optimal threshold value, which often turns out to be a manual process.

- λ-MAT, where λ is the shortest distance between the set of boundary points that correspond to the same point on the MAT [Chazal and Lieutier, 2005] (see Figure 2.18b). All point on the MAT for which this distance is higher than *lambda* contribute to the λ-MAT. Topology is preserved until the so-called weakest feature size is reached: the largest λ for which the (inner) MAT for a single object is still completely connected (compare Figures 2.15c and 2.15e). While seemingly trivial to implement in general, the λ-MAT has proven particularly popular in approximation methods that are based on the continuous distance transform [Chaussard et al., 2011; Hesselink et al., 2005].
- θ-MAT, where θ, or the separation angle, is the angle between two boundary points that correspond to the same point on the MAT (see Figure 2.18a). In case of more than two corresponding boundary points, the ones that result in the largest angle are chosen. The larger the separation angle of a given point on the MAT, the more stable that point is considered to be. It is generally not considered to be topology preserving. But it has been used by many

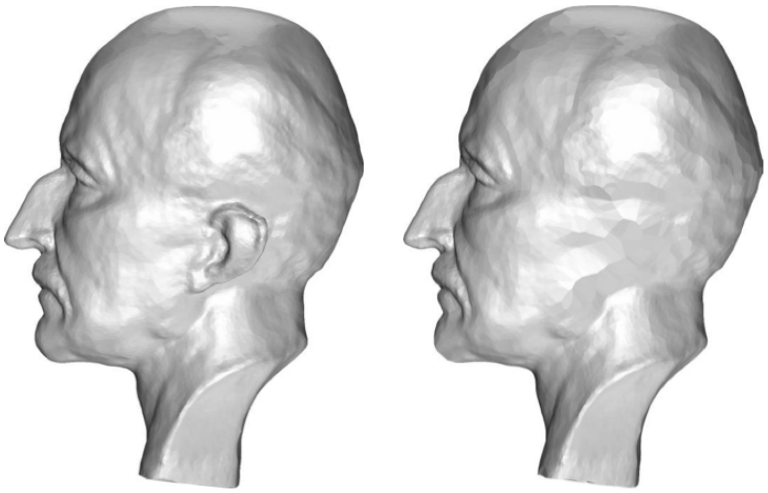
researchers [Amenta et al., 2001; Attali and Lachaud, 2001; Attali and Montanvert, 1996; Foskey et al., 2003]. Sometimes it is used together with other criteria [Dey and Zhao, 2004], or even in combination with topology-preserving constraints [Sud et al., 2007].

- The scale axis transform was mathematically introduced by Giesen et al. [2009], and then implemented by [Miklos et al., 2010]. It is different from other continuous pruning methods, because it can result in a *superset* of the original MAT, where the other methods alway result in a subset. The idea is to i) scale the radii of the medial balls of an initial MAT with a factor $s > 1$, ii) recompute the MAT of the union of scaled medial balls, and iii) re-scale the new medial balls with a factor of $1/s$. The resulting MAT has a simplified shape, because many relatively small medial balls will have disappeared in the second step. When comparing this method to the θ-MAT, Miklos et al. [2010] argue that it works in a more global fashion (because medial balls can disappear due to a merging with relatively far but significantly large balls) and is therefore of a higher quality. Topology is preserved up to a certain value of s, whereafter the topology might change due to the closing of holes, resulting in an object with a different genus. However, this problem is fixed in the work by Faraj et al. [2013], which improves and extends the scale axis transform. They use a progressive filtratation technique to compute the simplified MAT for all scales at once, this enables real-time exploring for the effects of different scale-threshold.
- The boundary potential of a point on $\mathcal{M}[X]$ is defined as the shortest distance between two corresponding boundary points over the surface of X (see Figure 2.18c). It was originally introduced in the 2D case by Ogniewicz and Ilg [1992] and a similar concept was used by Matuk et al. [2006]. Dey and Sun [2006] define essentially the same thing, but call it the *medial geodesic function*, and uses it to compute the curve skeleton: a topology equivalent subset of the MAT that has no area. Jalba et al. [2012] provide an efficient implementation and argue that it is a good measure because it takes into account the global shape of an object.

Part based pruning

means that the MAT is reduced based on the selective removal of entire MAT-sheets, usually while preserving topology.

- Tam and Heidrich [2003] introduce a three-step approach: i) decomposing the MAT into parts (the MAT sheets are cut where they meet), ii) assigning a significance value to the parts based on a) the number of triangles in the part, or b) the volume that would be removed from the input object as a result of pruning the part, and iii) performing an ordered pruning process that removes all parts with significance values within a specified range that do not alter the topology of the MAT. The order is such that outer parts are removed first. The number of iterations depends on the required level of simplification. Tam and Heidrich [2003] specifically mention that their aim is to perform feature-based pruning, i.e. to remove specific parts of a model as demonstrated in Figure 2.19.
- Sud et al. [2007] similarly decompose the MAT into parts and iteratively remove sheets (starting with the outer ones), but in contrast to Tam and Heidrich [2003], they assign a signifi-



(a) Before pruning (b) After pruning.

Figure 2.19: Using the part-based pruning method of Tam and Heidrich [2003], features can be selectively removed from a model.

cance value based on the highest separation angle (as defined earlier) in a sheet. Their primary reason to use part-based pruning is that it preserves topology and their objective is to identify a stable MAT.

Chapter 3

PhD research

3.1 Research question

This research ultimately aims to improve upon existing tools for DSM processing, such as thinning, and to explore new possiblities in the field of DSM analysis and generalization. More specifically, it investigates the feasibility of an approach based on the MAT to achieve feature identification and generalization in Digital Surface Models. Based on the literature reviewed in Chapter 2 that established that the MAT is a powerful tool for shape analysis, the hypotheses are that the 3D MAT gives the means to

1. define DSM features using the geometrical (medial ball radii and geometry of MAT points) and topological (branch connectivity and correspondence between surface and MAT points) information encoded in the MAT, and

2. enable a truly 3D analysis of modern DSMs, as opposed to the conventional 2.5D methods in GIS.

The main research question is:

- Can the MAT-concept be applied to analyze and generalize DSMs in a practically feasible manner that improves upon current methods?

In this context, DSM analysis means to:

1. identify *importance measures* that quantify the geometrical and topological configuration of the DSM, and

2. aggregate these measures to define application-specific features.

And DSM generalization means to apply this knowledge to either

1. reduce data-volume while preserving information content (mesh simplification), or

2. selectively remove application-specific shapes (shape simplification).

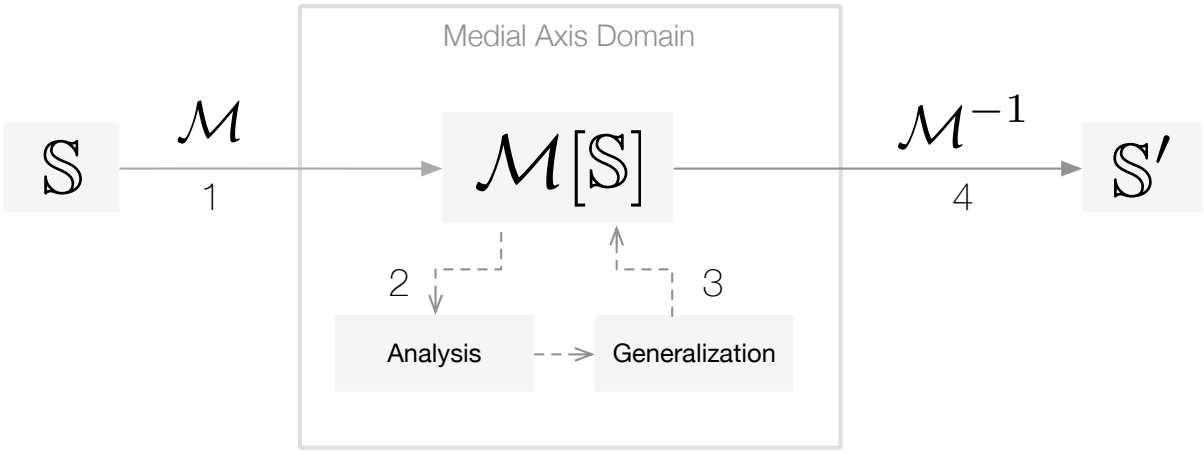


Figure 3.1: The proposed approach to generalize a DSM $\mathbb{S}$ using its MAT $\mathcal{M}[\mathbb{S}]$

Of course, this needs to be done effectively, meaning that the method should improve upon existing solutions and must be implementable in practice.

3.2 General approach

As stated, this research will focus on how to apply the *Medial Axis Transform* (MAT) concept to the analysis and generalization of DSMs. The expectation is that it is advantageous to perform certain operations, such as feature identification and feature-based generalization, on the MAT of the DSM (denoted as $\mathcal{M}[\mathbb{S}]$) rather than directly on the DSM $\mathbb{S}$ itself. The general workflow, as depicted in Figure 3.1, can be summarised into the following four steps:

1. compute (an approximation of) $\mathcal{M}[\mathbb{S}]$,
2. analyze $\mathcal{M}[\mathbb{S}]$ by exploiting its topological and geometrical properties to compute application specific signficance of parts and identify features,
3. generalize $\mathcal{M}[\mathbb{S}]$ based on the result of the analysis, and
4. reconstruct the generalized DSM $\mathbb{S}'$.

3.3 Methodology

In order to realize the four steps depicted in Figure 3.1, each step is subdivided in a number of tasks that are listed below. Figure 3.2 gives a more detailed overview of the intended workflow.

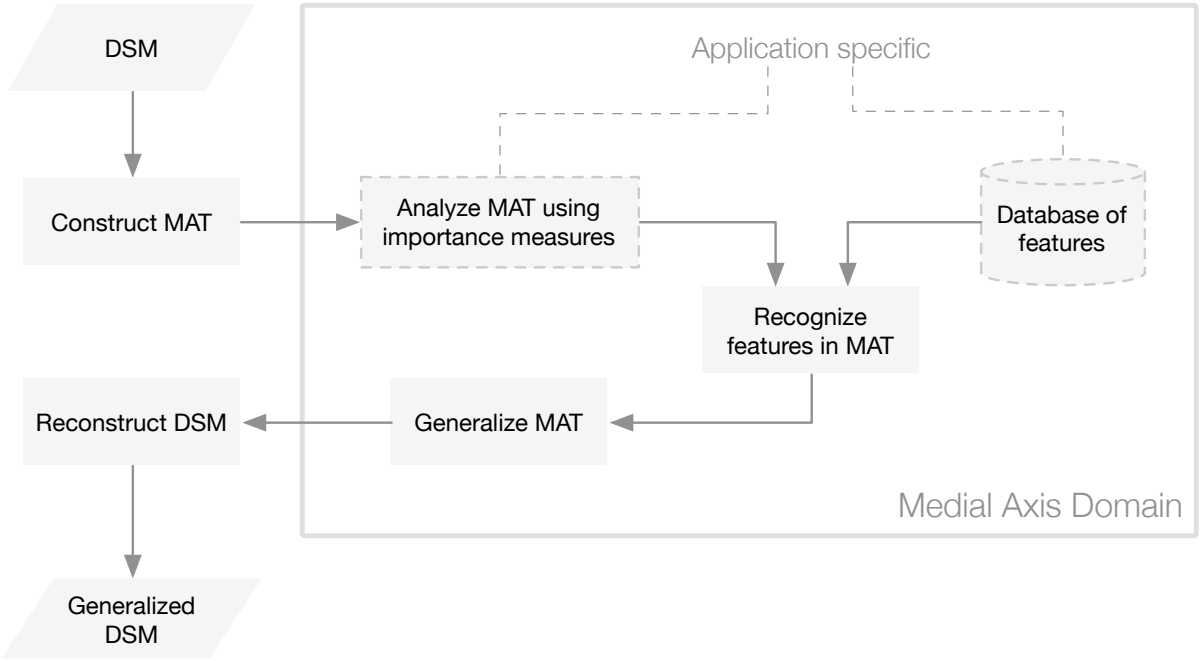


Figure 3.2: Workflow of the MAT approach

1. MAT construction
    a) Investigate MAT approximation algorithms. Identify and implement the one(s) most suited to DSMs.
    b) Investigate how to handle (DSM specific) flaws in input datasets such as noise and the non-closed nature of DSM surfaces. Implement and compare possible solutions.
    c) Investigate and implement suitable data-structures to represent and navigate in the MAT for DSMs.
2. MAT analysis for DSMs
    a) Investigate various importance measures (pruning) that can be defined on the MAT. Compare them and implement the most promising ones. If possible and necessary develop new importance measures.
    b) Investigate how application-dependent features (for case studies) can be defined using importance measures and how they can be extracted from the MAT.
    c) Investigate whether a hierarchy of features can be established from the MAT. This hierarchy would explicitly represent how different features are (geometrically) related, e.g. how different low-level features (e.g. houses) would relate to higher-level features (e.g. a housing block).

d) Determine how and to what extend the feature extraction process can be automated (likely to be case study dependent).

e) Investigate whether additional data (for instance 2D topographic maps/semantics) is needed to drive the feature extraction process and what is the best way to incorporate such (existing) data into the MAT-approach (case study dependent).

f) Investigate whether it is possible to do partial feature matching using the branch connectivity in the MAT.

3. MAT generalization for DSMs

a) Investigate how to use the MAT to reduce data-volume while preserving information content (similar to mesh simplification).

b) Investigate how to selectively remove application specific shapes (similar to shape simplification).

4. DSM reconstruction

a) Investigate how to effectively reconstruct (the altered part of) the DSM from the generalized MAT without introducing unnecessary alterations to the DSM.

5. Other issues

a) Investigate if and how the developed methods can be scaled up to massive datasets in terms of computation.

Since this research aims to deliver a practically feasible approach for the analysis and generalization of DSMs, part of the effort will be spent on actively developing MAT-based solutions for a number of relevant case studies. Ideally, these case studes will also benefit any involved parties in this project (see Section 4.3). These case studies can be considered to *drive* the research and will be developed as part of the tasks listed above. In addition, the following tasks are also identified.

6. Case studies

a) Find potential users and identify case studies that can be used to highlight the strenthgs of MAT based DSM analysis and/or generalization.

b) Demonstrate the feasibility of the MAT-based approach for every case study on a sample dataset.

c) Compare results to previously available solutions for every case study.

Finally, a software prototype that robustly implements the developed algorithms and data-structures will be developed.

Table 3.1: Initial validation plan

Part of research	Quantitive criteria
MAT construction	sensitivy to noise
MAT analysis and generalization	effectiveness of low-level feature recognition (e.g. though smart thinning)
	effectiveness of high-level feature recognition
DSM reconstruction	difference with original DSM
Scalability	running time dependence of input size in practice

3.4 Validation

In order to ensure this research is of a high quality, any (intermediate) results are to be validated thoroughly. This will be done both quantitively and qualitatively and independently for the different parts of this research as listed in Section 3.3. Since every part has a different aim that can be seen indepently from the rest of the research, the exact validation procedure depends on exactly what part is to be validated. It is expected that the validation methods will become more detailed and effective as the research progresses. However, an approximate validation plan can already be drafted. Some general methods of validation are:

1. Compare my results with results from relevant approaches in literature (e.g. some of the methods discussed in Chapter 2).
2. Study how errors (e.g. noise) in the input are propagated though the developed approach.
3. Compare results to those of existing approaches in practice (through case studies).

General qualitative approaches that will be applied to all parts of this research are:

1. Visual prototyping, i.e. to visualize intermediate steps in algorithms in effective ways so that it can be thoroughly understood what is going on.
2. Expert opinions. Interact with researchers from relevant fields through conferences and research visits and ask proffesional practioners (as listed in Section 4.3) for their opinion and suggestions.

Quantitative methods will be more specific to a certain part of the research, Table 3.1 lists a number of relevant quantative criteria for validation.

3.5 Scope/Priorities

1. Regarding sampling. It is obvious that the level of detail of observable features in a DSM strongly depends on the sampling density. Initially, the research will focus on ideal inputs that have high

sampling density and accuracy. Only after the methodology proves to be succesfull on these inputs, more flawed inputs will be considered

2. The goal of this research is not to perform surface reconstruction from point clouds. Therefore, and if needed, the assumption will be made that the input DSM is a surface.

3. Achieving scalability with massive inputs in terms of storage and computation time is not a primary objective. However, it will be considered during the development and implementation of the algorithms.

3.6 Risks

It is the nature of scientific research that the precise outcome can not be known in advance. Following is a list of potential obstacles for a succesfull outcome of parts of this research. These are challenges that ought te be solved and are foreseen to get significant attention in the context of this research. However, when they prove to be too difficult to solve (even after significant efforts have been made), an alternative is suggested.

No satisfactory importance measures can be established.
This would be an unfortunate outcome, since many tasks defined in the MAT analysis part depend on it. However, such an outcome may be mitigated by relying on data other than purely the geometry of the DSM (that defines the MAT). For example: radiometric data or any existing semantic information (case study dependent).

3.7 Case studies

The case studies form a crucial part of this research for the following reasons:

1. stimulating practical implementations of the designed algorithms,

2. motivation of research objectives,

3. providing sample datasets,

4. providing means to qualitatively and quantitatively validate the MAT-based approach by,comparing it to current methods for a specific (case study dependent) task,

5. exchanging knowledge with practitioners, and

6. demonstrating the general feasibility of the MAT-based approach.

The following three case studies have already been found.

Dyke profiles

The analysis of a dyke (needed for dyke monitoring and maintanance) requires a variety of data, such as information on the soil type, water levels and dyke-geometry. The latter, which is the focus of this case study, is commonly represented as a collection of 2D dyke-profiles, sampled at some regular interval along the dyke. The data preparation of these dyke profiles roughly consists of two steps:

1. deriving the actual dyke profiles (from AHN2), and
2. identifying a number of characteristic points in each dyke profile (see Figure 3.3).

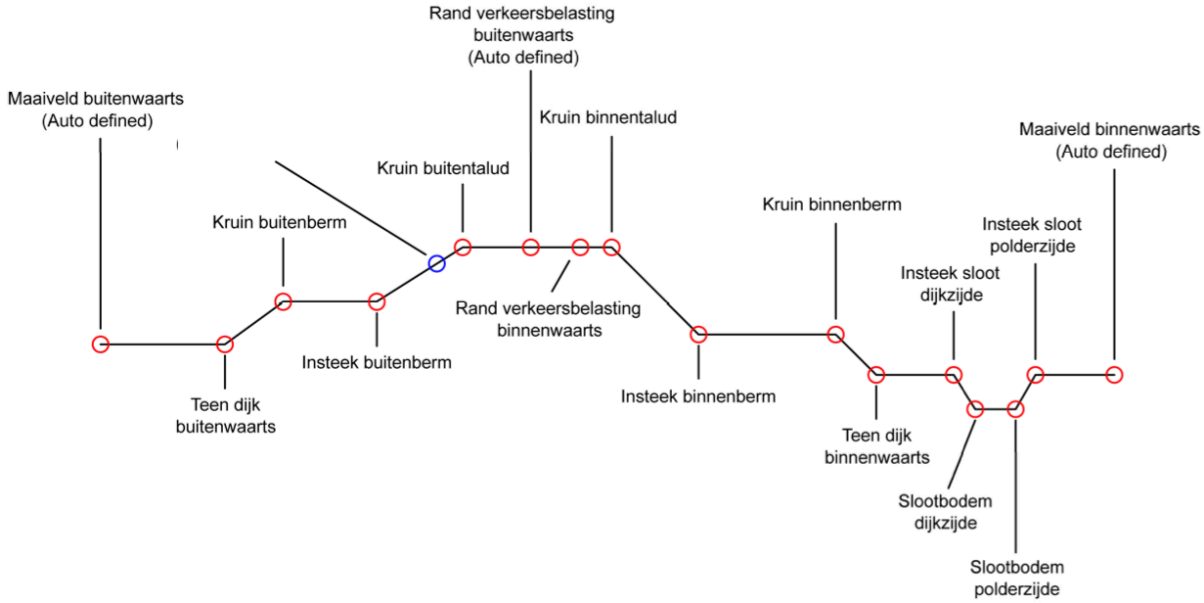


Figure 3.3: Characteristic points in a dyke profile required by the Dyke Analysis Module (DAM), a software platform for the automated analysis of dykes

The (automated) execution of both steps proves to be tedious in practice[1]. This is partly caused by the presence of noise in AHN2, but also by the presence of objects on the dike that serve no function with respect to water management (these are called 'niet waterkerende objecten'), such as traffic signs, trees and houses. That makes it difficult to develop a simple algorithm that can automatically trace the 'path' of a dike and extract profiles on the way. Furthermore, for the identification of the characteristic points in a dyke profile no automated method exists, and this is currently done completely manually.

This case study should investigate whether the MAT-based approach can be employed to

[1]For instance this Deltares software requires it `https://oss.deltares.nl/web/dam/home`

- identify dykes from a DSM,
- trace and extract dyke profiles along the dyke and identify critical (i.e. the weak spots in a dike) profiles,
- remove the so called 'niet waterkerende objecten' and,
- identify (semi-)automatically the charactistic points that define the dyke profile.

Parties involved: Grontmij, Waterschappen

Smart thinning

The Dutch Kadaster is currently working on a national topographic map in 3D [Oude Elberink et al., 2013]. For this purpose they intend to use AHN2 dataset to derive height information. To keep processing times on a reasonable level, the AHN2 dataset must be thinned in a preprocessing phase. Currently, they employ a simpe n^{th} point filter. Unfortunately, this thinning filter does not take into account local point configuration. As a result some areas end up with too few points to be useful, while in other areas the point density may still be decreased without greatly affecting the quality of the output.

This case study aims to investigate an adaptive thinning filter that considers local point configuration based on the MAT approach. The initial approach will attempt to apply a decimation based on the local feature size measure (that is the shortest distance from a surface point to the MAT) based on the ideas presented by Dey et al. [2001] and Ma et al. [2012]. However, before computing the local feature size for all surface points some form of noise detection is probably needed.

The goals of this case study are thus to

- investigate whether the local feature size is an effective measure for point decimation,
- investigate other importance measures if necessary, and
- investigate methods to detect and filter noisy points in the input DSM.

Parties involved: Kadaster

Simplification of roof geometry

Building geometries acquired through aerial LiDAR may be too detailed for some practical applications. Some of the clients of Safe Software require simplified roof geometries, for example without airconditioning units. The goals of this case study is to

- identify small object on top of roofs,
- remove these and compute a good surface representation for the simplified roof geometry (i.e. the gaps that result from the object removal must be filled).

Parties involved: Safe Software

3.8 Contributions

- A novel approach for the analysis of modern DSMs that is fully 3D.
- Further development of practically feasible algorithms to apply the MAT.
- Improved ability to extract features from DSM relevant to application.
- Improved DSM generalization.
- Robust software prototype that has been demonstrated to work for a number of case studies.

Chapter 4

Practical aspects

This chapter is mainly a summation of practical matters.

4.1 Technical matters

Following is a non-comprehensive list of potential software libraries and programs that may be used during the development of this project:

1. Scientific computation
    - numpy (Python) — `http://www.numpy.org`
    - scipy (Python) — `http://www.scipy.org`
    - CGAL — `http://www.cgal.org`
    - PCL — `http://www.pointclouds.org`
2. Data management
    - LASTools — `http://www.cs.unc.edu/~isenburg/lastools/`
    - GDAL/OGR — `http://www.gdal.org`
    - liblas/PDAL — `http://www.pdal.io`
    - Postgis — `http://postgis.net`
3. Visualisation and interaction
    - VRUI — `http://idav.ucdavis.edu/~okreylos/ResDev/Vrui/`
    - VTK — `http://www.vtk.org`
    - matplotlib (Python) — `http://matplotlib.org`

- D3 — http://d3js.org
- QGIS — http://www.qgis.org/en/site/
- LidarViewer — http://keckcaves.org/software/lidarviewer
- Meshlab — http://meshlab.sourceforge.net
- Blender — http://www.blender.org

4.2 Graduate School: courses and obligations

Every PhD candidate at TU Delft needs to fulfil the requirements of the university's Graduates School. This means that a sufficient number of activities (in 3 different categories) need to be performed. Table 4.1 gives an overview of the activities that have already been done, are currently being done, or are currently in planning. Note that this list is incomplete, additional activities will be added as the research progresses.

Table 4.1: Overview of courses and activities

Title	Type	Where	Status
Writing first journal article	Research skills	Marine Geodesy	completed
Writing a research proposal	Research skills	TU Delft	completed
Addressing small audience	Research skills	ICA workshop	completed
Assisting in laboratory course	Research skills	TU Delft	completed
Writing the first conference paper	Research skills	tbd	planned
Addressing major international audience	Research skills	tbd	planned
Poster presentation, major international audience	Research skills	tbd	planned
Teaching and active learning (C7.M1)	Research skills	TU Delft	planned
PhD Startup (C9.M1)	Transferable skills	TU Delft	completed
Scientific Writing in English (C13.M5)	Transferable skills	TU Delft	completed
Building a better brain (C11.M9)	Transferable skills	TU Delft	planned
What's you story? (C12.M7)	Transferable skills	TU Delft	planned
Geometric Algorithms	Discipline related	Utrecht University	in progress
Pattern Recognition	Discipline related	TU Delft	planned
Geometric Algorithms in the Field	Discipline related	Lorentz Center	planned

4.3 Involved parties

This project (dubbed *3DSM*), which is taking place within TU Delft, is financially 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). Table 4.2 lists the involved individuals from TU Delft and STW as well as a number of selected *users* from practice that have an interest in this research.

Table 4.2: People involved with the *3DSM* project

Name	Role	Organisation
Prof. Jantien Stoter	Promotor	TU Delft
Dr. Hugo Ledoux	Daily Supervisor	TU Delft
Dr. Roderik Lindenbergh	External reviewer	TU Delft
Ravi Peters, MSc	PhD student	TU Delft
Arend Zomer	Program officer	STW
René Visser	User	Rijkswaterstaat DVS
Niels van der Zon	User	Het waterschaphuis
Paul Beurskens	User	Grontmij
Joris Goos	User	Gemeente Rotterdam
Johan Cranenburgh	User	Waterschap scheldestromen
Marc Post	User	Kadaster
Benoit Frédéricque	User	Bentley
Kevin Wiebe	User	Safe Software

4.4 Planning

A rough planning of the main tasks that have already been performed or are to be performed for this research is given in Figure 4.1. More information on the core tasks (typeset in bold in the chart) is given in Section 3.3.

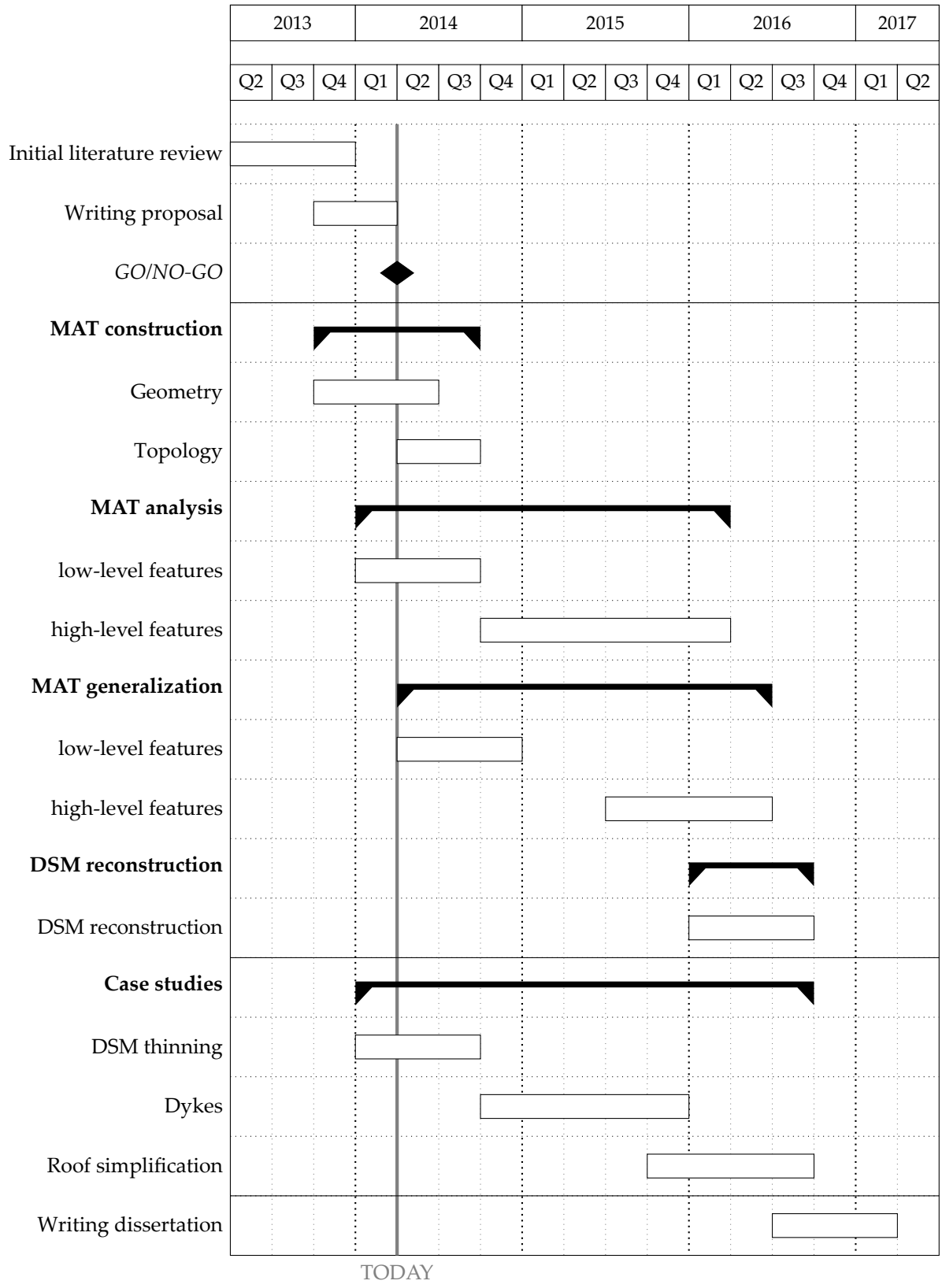


Figure 4.1: GANTT chart

Bibliography

AHN Website. URL http://www.ahn.nl.

Massive Point Clouds for eSciences Website. URL http://www.esciencecenter.nl/projects/project-portfolio/massive-point-clouds-for-esciences/.

P. K. Agarwal, L. Arge, and A. Danner. From point cloud to grid DEM: A scalable approach. In A. Reidl, W. Kainz, and G. Elmes, editors, *Progress in Spatial Data Handling—12th International Symposium on Spatial Data Handling*. Springer, 2006.

Oswin Aichholzer, Franz Aurenhammer, David Alberts, and Bernd Gärtner. *A novel type of skeleton for polygons*. Springer, 1996.

Nina Amenta and Sunghee Choi. Voronoi methods for 3d medial axis approximation. In *Medial Representations*, pages 223–239. Springer, 2008.

Nina Amenta and Ravi Krishna Kolluri. The medial axis of a union of balls. *Computational Geometry*, 20(1-2):25 – 37, 2001. ISSN 0925-7721. doi: 10.1016/S0925-7721(01)00033-5. URL http://www.sciencedirect.com/science/article/pii/S0925772101000335. <ce:title>Selected papers from the 12th Annual Canadian Conference on</ce:title>.

Nina Amenta, Marshall Bern, and David Eppstein. The crust and the β-skeleton: Combinatorial curve reconstruction. *Graphical Models and Image Processing*, 60(2):125 – 135, 1998a. ISSN 1077-3169. doi: 10.1006/gmip.1998.0465. URL http://www.sciencedirect.com/science/article/pii/S1077316998904658.

Nina Amenta, Marshall Bern, and Manolis Kamvysselis. A new voronoi-based surface reconstruction algorithm. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, pages 415–421, New York, NY, USA, 1998b. ACM. ISBN 0-89791-999-8. doi: 10.1145/280814.280947. URL http://doi.acm.org/10.1145/280814.280947.

Nina Amenta, Sunghee Choi, and Ravi Krishna Kolluri. The power crust. In *Proceedings of the sixth ACM symposium on Solid modeling and applications*, SMA '01, pages 249–266, New York, NY, USA, 2001. ACM.

Lars Arge, Kasper Green Larsen, Thomas Mølhave, and Freek van Walderveen. Cleaning massive sonar point clouds. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '10, pages 152–161, New York, NY, USA, 2010. ACM.

Dominique Attali and J.-O. Lachaud. Delaunay conforming iso-surface, skeleton extraction and noise removal. *Computational Geometry*, 19(2–3):175 – 189, 2001. ISSN 0925-7721. doi: http://dx.doi.org/10.1016/S0925-7721(01)00019-0. URL http://www.sciencedirect.com/science/article/pii/S0925772101000190. <ce:title>Combinatorial Curves and Surfaces</ce:title>.

Dominique Attali and Annick Montanvert. Modeling noise for a better simplification of skeletons. In *Image Processing, 1996. Proceedings., International Conference on*, volume 3, pages 13–16. IEEE, 1996.

Dominique Attali and Annick Montanvert. Computing and simplifying 2d and 3d continuous skeletons. *Computer Vision and Image Understanding*, 67(3):261 – 273, 1997. ISSN 1077-3142. doi: 10.1006/cviu.1997.0536. URL http://www.sciencedirect.com/science/article/pii/S1077314297905361.

Dominique Attali, Jean-Daniel Boissonnat, and Herbert Edelsbrunner. Stability and computation of medial axes-a state-of-the-art report. *Mathematical foundations of scientific visualization, computer graphics, and massive data exploration*, pages 109–125, 2009.

Gill Barequet, David Eppstein, Michael T Goodrich, and Amir Vaxman. Straight skeletons of three-dimensional polyhedra. In *Algorithms-ESA 2008*, pages 148–160. Springer, 2008.

Silvia Biasotti, Dominique Attali, Jean-Daniel Boissonnat, Herbert Edelsbrunner, Gershon Elber, Michela Mortara, GabriellaSanniti Baja, Michela Spagnuolo, Mirela Tanase, and Remco Veltkamp. Skeletal structures. In Leila Floriani and Michela Spagnuolo, editors, *Shape Analysis and Structuring*, Mathematics and Visualization, pages 145–183. Springer Berlin Heidelberg, 2008. ISBN 978-3-540-33264-0.

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

Harry Blum. Biological shape and visual science (part i). *Journal of Theoretical Biology*, 38(2):205 – 287, 1973. ISSN 0022-5193. doi: http://dx.doi.org/10.1016/0022-5193(73)90175-6. URL http://www.sciencedirect.com/science/article/pii/0022519373901756.

Gunilla Borgefors, Ingela Nyström, and Gabriella Sanniti di Baja. Discrete skeletons from distance transforms in 2d and 3d. In *Medial Representations*, pages 155–190. Springer, 2008.

Jonathan W Brandt and V Ralph Algazi. Continuous skeleton computation by voronoi diagram. *CVGIP: Image Understanding*, 55(3):329–338, 1992.

Ming-Ching Chang. *The Medial Scaffold for 3D Shape Modeling and Recognition*. Ph.D. dissertation, Division of Engineering, Brown University, Providence, RI, 02912 USA, Sep. 2008.

John Chaussard, Michel Couprie, and Hugues Talbot. Robust skeletonization using the discrete λ-medial axis. *Pattern Recognition Letters*, 32(9):1384 – 1394, 2011. ISSN 0167-8655. doi: 10.1016/j.patrec.2010.09.002. URL http://www.sciencedirect.com/science/article/pii/S0167865510003004.

Frédéric Chazal and André Lieutier. The lambda-medial axis. *Graphical Models*, 67(4):304–331, 2005.

Catalin Constantin, Shawn Brown, and Jack Snoeyink. Implementing streaming simplification for large labeled meshes. In *Proceedings Algorithm Engineering and Experiments (ALENEX10)*, pages 149–158, Austin, Texas, USA, 2010.

Clayton Crawford. Assessing the use of point thinning on airborne lidar for dem production. Presentation slides, European Lidar Mapping Forum 2013, Amsterdam, nov. 2013.

Tamal K Dey and Jian Sun. Defining and computing curve-skeletons with medial geodesic function. In *Symposium on Geometry Processing*, pages 143–152, 2006.

Tamal K Dey and Wulue Zhao. Approximate medial axis as a voronoi subcomplex. *Computer-Aided Design*, 36(2):195–202, 2004.

Tamal K. Dey, Joachim Giesen, and James Hudson. Decimating samples for mesh simplification. In *Proc. 13th Canadian Conf. Comput. Geom*, pages 85–88, 2001.

Herbert Edelsbrunner, John Harer, and Afra Zomorodian. Hierarchical morse complexes for piecewise linear 2-manifolds. In *Proceedings of the seventeenth annual symposium on Computational geometry*, SCG '01, pages 70–79, New York, NY, USA, 2001. ACM. ISBN 1-58113-357-X. doi: 10.1145/378583.378626. URL http://doi.acm.org/10.1145/378583.378626.

Ian S Evans. Geomorphometry and landform mapping: What is a landform? *Geomorphology*, 137(1): 94–106, 2012.

Noura Faraj, Jean-Marc Thiery, and Tamy Boubekeur. Progressive medial axis filtration. *ACM SIGGRAPH 2013, Technical Briefs*, 2013.

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

Mark Foskey, Ming C. Lin, and Dinesh Manocha. Efficient computation of a simplified medial axis. In *Proceedings of the eighth ACM symposium on Solid modeling and applications*, SM '03, pages 96–107, New York, NY, USA, 2003. ACM.

M. Garland. Multiresolution modeling: Survey & future opportunities. *State of the Art Report*, pages 111–131, 1999.

M. Garland and P.S. Heckbert. *Fast polygonal approximation of terrains and height fields*. School of Computer Science, Carnegie Mellon University, 1995.

Michael Garland and P.S. Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '97, pages 209–216, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

John M Gauch and Stephen M Pizer. The intensity axis of symmetry and its application to image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(8):753–770, 1993.

Joachim Giesen, Balint Miklos, Mark Pauly, and Camille Wormser. The scale axis transform. In *Proceedings of the 25th annual symposium on Computational geometry*, pages 106–115. ACM, 2009.

C. Gold and J. Snoeyink. A one-step crust and skeleton extraction algorithm. *Algorithmica*, 30(2):144–163, 2001. ISSN 0178-4617. doi: 10.1007/s00453-001-0014-x. URL `http://dx.doi.org/10.1007/s00453-001-0014-x`.

Michael F. Goodchild. Geographical data modeling. *Computers & Geosciences*, 18(4):401–408, 1992.

Norbert Haala and Martin Kada. An update on automatic 3d building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6):570–580, 2010.

P.S. Heckbert and M. Garland. Survey of polygonal surface simplification algorithms. Technical report, DTIC Document, 1997.

WimH. Hesselink, Menno Visser, and JosB.T.M. Roerdink. Euclidean skeletons of 3d data sets in linear time by the integer medial axis transform. In Christian Ronse, Laurent Najman, and Etienne Decencière, editors, *Mathematical Morphology: 40 Years On*, volume 30 of *Computational Imaging and Vision*, pages 259–268. Springer Netherlands, 2005. ISBN 978-1-4020-3442-8. doi: 10.1007/1-4020-3443-1_23. URL `http://dx.doi.org/10.1007/1-4020-3443-1_23`.

H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. *Surface reconstruction from unorganized points*. PhD thesis, University of Washington, 1994.

Martin Isenburg and P. Lindstrom. Streaming meshes. In *Visualization, 2005. VIS 05. IEEE*, pages 231 – 238, oct. 2005. doi: 10.1109/VISUAL.2005.1532800.

Martin Isenburg, Yuanxin Liu, Jonathan Richard Shewchuk, and Jack Snoeyink. Streaming computation of Delaunay triangulations. *ACM Transactions on Graphics*, 25(3):1049–1056, 2006a.

Martin Isenburg, Yuanxin Liu, Jonathan Richard Shewchuk, Jack Snoeyink, and Tim Thirion. Generating raster DEM from mass points via TIN streaming. In *Geographic Information Science—GIScience 2006*, volume 4197 of *Lecture Notes in Computer Science*, pages 186–198, Münster, Germany, 2006b.

Martin Isenburg, Yuanxin Liu, and Jack Snoeyink. Streaming extraction of elevation contours from LIDAR points. Not published, available at `http://www.cs.unc.edu/~isenburg/papers/ils-lidar2iso-06.pdf`, 2006c.

Andrei C. Jalba, Jacek Kustra, and Alexandru C. Telea. Surface and curve skeletonization of large 3d models on the gpu. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(6):1495–1508, June 2012.

Takeo Kanade. Extracting topographic terrain features from elevation maps. 1994.

Youngihn Kho and Michael Garland. User-guided simplification. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, I3D '03, pages 123–126, New York, NY, USA, 2003. ACM. ISBN 1-58113-645-5. doi: http://doi.acm.org/10.1145/641480.641504. URL http://doi.acm.org/10.1145/641480.641504.

Oliver Kreylos, Gerald Bawden, and Louise Kellogg. Immersive visualization and analysis of lidar data. *Advances in Visual Computing*, pages 846–855, 2008.

Florent Lafarge and Clément Mallet. Creating large-scale city models from 3d-point clouds: a robust approach with hybrid representation. *International journal of computer vision*, 99(1):69–85, 2012.

J. Lee. *A drop heuristic conversion method for extracting irregular network for digital elevation models*. ASPRS/ACSM, 1989.

Zhilin Li, Qing Zhu, and Chris Gold. *Digital terrain modeling: principles and methodology*. CRC press, 2010.

Jaehwan Ma, Sang Won Bae, and Sunghee Choi. 3d medial axis point approximation using nearest neighbors and the normal field. *The Visual Computer*, 28(1):7–19, 2012.

Paola Magillo, Emanuele Danovaro, Leila De Floriani, Laura Papaleo, and Maria Vitali. Extracting terrain morphology-a new algorithm and a comparative evaluation. In *GRAPP (GM/R)*, pages 13–20, 2007.

Paola Magillo, Emanuele Danovaro, Leila Floriani, Laura Papaleo, and Maria Vitali. A discrete approach to compute terrain morphology. *Computer Vision and Computer Graphics. Theory and Applications*, pages 13–26, 2009.

K. Matuk, C. Gold, and Z. Li. Skeleton based contour line generalization. *Progress in Spatial Data Handling*, pages 643–658, 2006.

Krzysztof Matuk. *Feature-based terrain model simplification*. PhD thesis, Hong Kong Polytechnic University, 2006.

Balint Miklos, Joachim Giesen, and Mark Pauly. Discrete scale axis representations for 3d geometry. *ACM Trans. Graph.*, 29:101:1–101:10, July 2010. ISSN 0730-0301. doi: http://doi.acm.org/10.1145/1778765.1778838. URL http://doi.acm.org/10.1145/1778765.1778838.

P. Musialski, P. Wonka, D. G. Aliaga, M. Wimmer, L. van Gool, and W. Purgathofer. A survey of urban reconstruction. *Computer Graphics Forum*, pages n/a–n/a, 2013. ISSN 1467-8659. doi: 10.1111/cgf.12077. URL http://dx.doi.org/10.1111/cgf.12077.

R Ogniewicz and M Ilg. Voronoi skeletons: Theory and applications. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pages 63–69. IEEE, 1992.

Sander Oude Elberink, Jantien Stoter, Hugo Ledoux, and Tom Commandeur. Generation and dissemination of a national virtual 3d city and landscape model for the netherlands. *Photogrammetric engineering and remote sensing*, 79(2):147–158, 2013.

Friso Penninga. Towards 3d topography using a feature-based integrated tin/ten model. In *AGILE*, pages 26–28, 2005.

RJ Pike, IS Evans, and T Hengl. Geomorphometry: A brief guide. *Developments in Soil Science*, 33:3–30, 2009.

Doron Shaked and Alfred M. Bruckstein. Pruning medial axes. *Computer Vision and Image Understanding*, 69(2):156 – 169, 1998. ISSN 1077-3142. doi: 10.1006/cviu.1997.0598. URL `http://www.sciencedirect.com/science/article/pii/S1077314297905981`.

Kaleem Siddiqi and Stephen M Pizer. *Medial representations: mathematics, algorithms and applications*, volume 37. Springer, 2008.

Gregory I Snyder. The benefits of improved national elevation data. *Photogrammetric Engineering and Remote Sensing*, 79(2), 2013.

André Sobiecki, Haluk C Yasan, Andrei C Jalba, and Alexandru C Telea. Qualitative comparison of contraction-based curve skeletonization methods. In *Mathematical Morphology and Its Applications to Signal and Image Processing*, pages 425–439. Springer, 2013.

Avneesh Sud, Mark Foskey, and Dinesh Manocha. Homotopy-preserving medial axis simplification. *International Journal of Computational Geometry & Applications*, 17(05):423–451, 2007.

Roger Tam and Wolfgang Heidrich. Feature-preserving medial axis noise removal. In *Computer Vision—ECCV 2002*, pages 672–686. Springer, 2002.

Roger Tam and Wolfgang Heidrich. Shape simplification based on the medial axis transform. In *Visualization, 2003. VIS 2003. IEEE*, pages 481–488. IEEE, 2003.

Mirela Tănase and Remco C Veltkamp. A straight skeleton approximating the medial axis. In *Algorithms–ESA 2004*, pages 809–821. Springer, 2004.

JohanW.H. Tangelder and RemcoC. Veltkamp. A survey of content based 3d shape retrieval methods. *Multimedia Tools and Applications*, 39(3):441–471, 2008. ISSN 1380-7501. doi: 10.1007/s11042-007-0181-0. URL `http://dx.doi.org/10.1007/s11042-007-0181-0`.

M. van Kreveld. Digital elevation models and tin algorithms. *Algorithmic foundations of Geographic Information Systems*, pages 37–78, 1997.

M. Van Kreveld, R. Van Oostrum, C. Bajaj, V. Pascucci, and D. Schikore. Contour trees and small seed sets for isosurface traversal. In *Proceedings of the thirteenth annual symposium on Computational geometry*, pages 212–220. ACM, 1997.

J. D. Wood. *The Geomorphological Characterisation of Digital Elevation Models*. PhD thesis, University of Leicester, UK, 1996.

Qiming Zhou and A-Xing Zhu. The recent advancement in digital terrain analysis and modeling. *International Journal of Geographical Information Science*, 27(7):1269–1271, 2013. doi: 10.1080/13658816.2013.794281. URL `http://www.tandfonline.com/doi/abs/10.1080/13658816.2013.794281`.

Reports published before in this series

1. GISt Report No. 1, Oosterom, P.J. van, Research issues in integrated querying of geometric and thematic cadastral information (1), Delft University of Technology, Rapport aan Concernstaf Kadaster, Delft 2000, 29 p.p.
2. GISt Report No. 2, Stoter, J.E., Considerations for a 3D Cadastre, Delft University of Technology, Rapport aan Concernstaf Kadaster, Delft 2000, 30.p.
3. GISt Report No. 3, Fendel, E.M. en A.B. Smits (eds.), Java GIS Seminar, Opening GDMC, Delft 15 November 2000, Delft University of Technology, GISt. No. 3, 25 p.p.
4. GISt Report No. 4, Oosterom, P.J.M. van, Research issues in integrated querying of geometric and thematic cadastral information (2), Delft University of Technology, Rapport aan Concernstaf Kadaster, Delft 2000, 29 p.p.
5. GISt Report No. 5, Oosterom, P.J.M. van, C.W. Quak, J.E. Stoter, T.P.M. Tijssen en M.E. de Vries, Objectgerichtheid TOP10vector: Achtergrond en commentaar op de gebruikersspecificaties en het conceptuele gegevensmodel, Rapport aan Topografische Dienst Nederland, E.M. Fendel (eds.), Delft University of Technology, Delft 2000, 18 p.p.
6. GISt Report No. 6, Quak, C.W., An implementation of a classification algorithm for houses, Rapport aan Concernstaf Kadaster, Delft 2001, 13.p.
7. GISt Report No. 7, Tijssen, T.P.M., C.W. Quak and P.J.M. van Oosterom, Spatial DBMS testing with data from the Cadastre and TNO NITG, Delft 2001, 119 p.
8. GISt Report No. 8, Vries, M.E. de en E. Verbree, Internet GIS met ArcIMS, Delft 2001, 38 p.
9. GISt Report No. 9, Vries, M.E. de, T.P.M. Tijssen, J.E. Stoter, C.W. Quak and P.J.M. van Oosterom, The GML prototype of the new TOP10vector object model, Report for the Topographic Service, Delft 2001, 132 p.
10. GISt Report No. 10, Stoter, J.E., Nauwkeurig bepalen van grondverzet op basis van CAD ontgravingsprofielen en GIS, een haalbaarheidsstudie, Rapport aan de Bouwdienst van Rijkswaterstaat, Delft 2001, 23 p.
11. GISt Report No. 11, Geo DBMS, De basis van GIS-toepassingen, KvAG/AGGN Themamiddag, 14 november 2001, J. Flim (eds.), Delft 2001, 37 p.
12. GISt Report No. 12, Vries, M.E. de, T.P.M. Tijssen, J.E. Stoter, C.W. Quak and P.J.M. van Oosterom, The second GML prototype of the new TOP10vector object model, Report for the Topographic Service, Delft 2002, Part 1, Main text, 63 p. and Part 2, Appendices B and C, 85 p.
13. GISt Report No. 13, Vries, M.E. de, T.P.M. Tijssen en P.J.M. van Oosterom, Comparing the storage of Shell data in Oracle spatial and in Oracle/ArcSDE compressed binary format, Delft 2002, .72 p. (Confidential)
14. GISt Report No. 14, Stoter, J.E., 3D Cadastre, Progress Report, Report to Concernstaf Kadaster, Delft 2002, 16 p.
15. GISt Report No. 15, Zlatanova, S., Research Project on the Usability of Oracle Spatial within the RWS Organisation, Detailed Project Plan (MD-NR. 3215), Report to Meetkundige Dienst – Rijkswaterstaat, Delft 2002, 13 p.
16. GISt Report No. 16, Verbree, E., Driedimensionale Topografische Terreinmodellering op basis van Tetraëder Netwerken: Top10-3D, Report aan Topografische Dienst Nederland, Delft 2002, 15 p.
17. GISt Report No. 17, Zlatanova, S. Augmented Reality Technology, Report to SURFnet bv, Delft 2002, 72 p.
18. GISt Report No. 18, Vries, M.E. de, Ontsluiting van Geo-informatie via netwerken, Plan van aanpak, Delft 2002, 17p.
19. GISt Report No. 19, Tijssen, T.P.M., Testing Informix DBMS with spatial data from the cadastre, Delft 2002, 62 p.
20. GISt Report No. 20, Oosterom, P.J.M. van, Vision for the next decade of GIS technology, A research agenda for the TU Delft the Netherlands, Delft 2003, 55 p.
21. GISt Report No. 21, Zlatanova, S., T.P.M. Tijssen, P.J.M. van Oosterom and C.W. Quak, Research on usability of Oracle Spatial within the RWS organisation, (AGI-GAG-2003-21), Report to Meetkundige Dienst – Rijkswaterstaat, Delft 2003, 74 p.
22. GISt Report No. 22, Verbree, E., Kartografische hoogtevoorstelling TOP10vector, Report aan Topografische Dienst Nederland, Delft 2003, 28 p.

23. GISt Report No. 23, Tijssen, T.P.M., M.E. de Vries and P.J.M. van Oosterom, Comparing the storage of Shell data in Oracle SDO_Geometry version 9i and version 10g Beta 2 (in the context of ArcGIS 8.3), Delft 2003, 20 p. (Confidential)
24. GISt Report No. 24, Stoter, J.E., 3D aspects of property transactions: Comparison of registration of 3D properties in the Netherlands and Denmark, Report on the short-term scientific mission in the CIST – G9 framework at the Department of Development and Planning, Center of 3D geo-information, Aalborg, Denmark, Delft 2003, 22 p.
25. GISt Report No. 25, Verbree, E., Comparison Gridding with ArcGIS 8.2 versus CPS/3, Report to Shell International Exploration and Production B.V., Delft 2004, 14 p. (confidential).
26. GISt Report No. 26, Penninga, F., Oracle 10g Topology, Testing Oracle 10g Topology with cadastral data, Delft 2004, 48 p.
27. GISt Report No. 27, Penninga, F., 3D Topography, Realization of a three dimensional topographic terrain representation in a feature-based integrated TIN/TEN model, Delft 2004, 27 p.
28. GISt Report No. 28, Penninga, F., Kartografische hoogtevoorstelling binnen TOP10NL, Inventarisatie mogelijkheden op basis van TOP10NL uitgebreid met een Digitaal Hoogtemodel, Delft 2004, 29 p.
29. GISt Report No. 29, Verbree, E. en S.Zlatanova, 3D-Modeling with respect to boundary representations within geo-DBMS, Delft 2004, 30 p.
30. GISt Report No. 30, Penninga, F., Introductie van de 3e dimensie in de TOP10NL; Voorstel voor een onderzoekstraject naar het stapsgewijs introduceren van 3D data in de TOP10NL, Delft 2005, 25 p.
31. GISt Report No. 31, P. van Asperen, M. Grothe, S. Zlatanova, M. de Vries, T. Tijssen, P. van Oosterom and A. Kabamba, Specificatie datamodel Beheerkaart Nat, RWS-AGI report/GIST Report, Delft, 2005, 130 p.
32. GISt Report No. 32, E.M. Fendel, Looking back at Gi4DM, Delft 2005, 22 p.
33. GISt Report No. 33, P. van Oosterom, T. Tijssen and F. Penninga, Topology Storage and the Use in the context of consistent data management, Delft 2005, 35 p.
34. GISt Report No. 34, E. Verbree en F. Penninga, RGI 3D Topo - DP 1-1, Inventarisatie huidige toegankelijkheid, gebruik en mogelijke toepassingen 3D topografische informatie en systemen, 3D Topo Report No. RGI-011-01/GISt Report No. 34, Delft -2005, 29 p.
35. GISt Report No. 35, E. Verbree, F. Penninga en S. Zlatanova, Datamodellering en datastructurering voor 3D topografie, 3D Topo Report No. RGI-011-02/GISt Report No. 35, Delft 2005, 44 p.
36. GISt Report No. 36, W. Looijen, M. Uitentuis en P. Bange, RGI-026: LBS-24-7, Tussenrapportage DP-1: Gebruikerswensen LBS onder redactie van E. Verbree en E. Fendel, RGI LBS-026-01/GISt Rapport No. 36, Delft 2005, 21 p.
37. GISt Report No. 37, C. van Strien, W. Looijen, P. Bange, A. Wilcsinszky, J. Steenbruggen en E. Verbree, RGI-026: LBS-24-7, Tussenrapportage DP-2: Inventarisatie geo-informatie en -services onder redactie van E. Verbree en E. Fendel, RGI LBS-026-02/GISt Rapport No. 37, Delft 2005, 21 p.
38. GISt Report No. 38, E. Verbree, S. Zlatanova en E. Wisse, RGI-026: LBS-24-7, Tussenrapportage DP-3: Specifieke wensen en eisen op het gebied van plaatsbepaling, privacy en beeldvorming, onder redactie van E. Verbree en E. Fendel, RGI LBS-026-03/GISt Rapport No. 38, Delft 2005, 15 p.
39. GISt Report No. 39, E. Verbree, E. Fendel, M. Uitentuis, P. Bange, W. Looijen, C. van Strien, E. Wisse en A. Wilcsinszky en E. Verbree, RGI-026: LBS-24-7, Eindrapportage DP-4: Workshop 28-07-2005 Geo-informatie voor politie, brandweer en hulpverlening ter plaatse, RGI LBS-026-04/GISt Rapport No. 39, Delft 2005, 18 p.
40. GISt Report No. 40, P.J.M. van Oosterom, F. Penninga and M.E. de Vries, Trendrapport GIS, GISt Report No. 40 / RWS Report AGI-2005-GAB-01, Delft, 2005, 48 p.
41. GISt Report No. 41, R. Thompson, Proof of Assertions in the Investigation of the Regular Polytope, GISt Report No. 41 / NRM-ISS090, Delft, 2005, 44 p.
42. GISt Report No. 42, F. Penninga and P. van Oosterom, Kabel- en leidingnetwerken in de kadastrale registratie (in Dutch) GISt Report No. 42, Delft, 2006, 38 p.
43. GISt Report No. 43, F. Penninga and P.J.M. van Oosterom, Editing Features in a TEN-based DBMS approach for 3D Topographic Data Modelling, Technical Report, Delft, 2006, 21 p.
44. GISt Report No. 44, M.E. de Vries, Open source clients voor UMN MapServer: PHP/Mapscript, JavaScript, Flash of Google (in Dutch), Delft, 2007, 13 p.
45. GISt Report No. 45, W. Tegtmeier, Harmonization of geo-information related to the lifecycle of civil engineering objects – with focus on uncertainty and quality of surveyed data and derived real world representations, Delft, 2007, 40 p.
46. GISt Report No. 46, W. Xu, Geo-information and formal semantics for disaster management, Delft, 2007, 31 p.
47. GISt Report No. 47, E. Verbree and E.M. Fendel, GIS technology - Trend Report, Delft, 2007, 30 p.
48. GISt Report No. 48, B.M. Meijers, Variable-Scale Geo-Information, Delft, 2008, 30 p.
49. GISt Report No. 48, Maja Bitenc, Kajsa Dahlberg, Fatih Doner, Bas van Goort, Kai Lin,Yi Yin, Xiaoyu Yuan and Sisi Zlatanova, Utilty Registration, Delft, 2008, 35 p.
50. GISt Report No 50, T.P.M. Tijssen en S. Zlatanova, Oracle Spatial 11g en ArcGIS 9.2 voor het beheer van puntenwolken (Confidential), Delft, 2008, 16 p.
51. GISt Report No. 51, S. Zlatanova, Geo-information for Crisis Management, Delft, 2008, 24 p.

52. GISt Report No. 52, P.J.M. van Oosterom, INSPIRE activiteiten in het jaar 2008 (partly in Dutch), Delft, 2009, 142 p.
53. GISt Report No. 53, P.J.M. van Oosterom with input of and feedback by Rod Thompson and Steve Huch (Department of Environment and Resource Management, Queensland Government), Delft, 2010, 60 p.
54. GISt Report No. 54, A. Dilo and S. Zlatanova, Data modeling for emergency response, Delft, 2010, 74 p.
55. GISt Report No. 55, Liu Liu, 3D indoor "door-to-door" navigation approach to support first responders in emergency response - PhD Research Proposal, Delft, 2011, 47 p.
56. GISt Report No. 56, Md. Nazmul Alam, Shadow effect on 3D City Modelling for Photovoltaic Cells - PhD Proposal, Delft, 2011, 39 p.
57. GISt Report No. 57, G.A.K. Arroyo Ohori, Realising the Foundations of a Higher Dimensional GIS: A Study of Higher Dimensional Data Models, Data Structures and Operations - PhD Research Proposal, Delft, 2011, 68 p.
58. GISt Report No. 58, Zhiyong Wang, Integrating Spatio-Temporal Data into Agent-Based Simulation for Emergency Navigation Support - PhD Research Proposal, Delft, 2012, 49 p.
59. GISt Report No. 59, Theo Tijssen, Wilko Quak and Peter van Oosterom, Geo-DBMS als standard bouwsteen voor Rijkswaterstaat (in Dutch), Delft, 2012, 167 p.
60. GISt Report No. 60, Amin Mobasheri, Designing formal semantics of geo-information for disaster response - PhD Research Proposal, Delft, 2012, 61 p.
61. GISt Report No. 61, Simeon Nedkov, Crowdsourced WebGIS for routing applications in disaster management situations, Delft, 2012, 31 p.
62. GISt Report No. 62, Filip Biljecki, The concept of level of detail in 3D city models – PhD Research Proposal, Delft, 2013, 58 p.
63. GISt Report No. 63, Theo Tijssen & Wilko Quak, GISt activiteiten voor het GeoValley project – Projectnummer: GBP / 21F.005, Delft, 2013, 39 p.
64. GISt Report No. 64, Radan Šuba, Content of Variable-scale Maps – PhD Proposal, Delft, 2013, 36 p.


ZUIDPOLDER
VAN DELFGAUW
MOLENSLOOT
D
P
POLDER