

MSc Graduation Plan (P2) in Geomatics

Constructing a digital 3D road network for The Netherlands

Kristof Kenesei

Student 5142334

K.Kenessei@student.tudelft.nl

Supervisors: 1st – Ravi Peters
2nd – Jantien Stoter

P2 Date: 2020-01-19



January 10, 2021

1 Introduction

Constructing and maintaining an up-to-date graph-like road network on the national level has a range of firmly established uses. Owing to its structure, it can be used efficiently for modelling and simulation purposes, such as for traffic flow simulations, passenger transport modelling, construction and upgrade impact modelling (to pinpoint optimal locations and types of investment), and traffic noise load modelling (Bell and Lida [1997]; Zhu and Li [2007]; Zhang [2011]; Durán Fernández and Santos [2014]; Peng et al. [2020]). It can also be used for navigation; a graph-like road network representation is at the heart of most road navigation services (Yue et al. [2008]). Combined with other data sets, we can mention an even wider range of use cases: complemented by ecological statistics and models, it can offer insight into the impact of the presence of roads, and planned road construction on the flora and fauna in their vicinity. Or to mention logistical benefits, a good model representing the road network in a digital format may be used as a shared working space when aggregating geospatial data relating to road infrastructure from various sources. It makes it possible for geographical road locations, topographical relationships, and arbitrary semantic information to reside in the same network-type data model, making analysis techniques more straightforward, enforcing consistency and saving effort for data providers who would otherwise all need to maintain their own road model (Ekpenyong et al. [2007]).

One may remark that a two-dimensional representation with approximate geographical locations may suffice for many of these purposes, topology being the main concern in network analysis. For instance, GNSS navigation software often use snapping methods to ensure that the navigating vehicle always traverses the road graph – ensuring that even for imperfect road locations and positioning, the navigation remains continuous (Fouque and Bonnifait [2008]; Chen and Hsu [2020]). Traffic flow simulations are primarily concerned with traffic loads, road properties, and how roads are subdivided by intersections, among many other aspects. *Mostly*, they are not concerned with the exact geographical locations of roads – as long as the lengths of the roads are correct, any geographical permutation will yield invariant results (Thomson and Richardson [1995]).

However, some applications are concerned with the road network in the context of its surroundings, which makes the accuracy of its georeferencing a relevant topic. Noise modelling is one such example, because the propagation of road noise is sensitive to the height of the road itself as well as to the immediate surroundings of the roads, requiring exact geographical locations of non-road objects surrounding the road (such as buildings or noise barriers), and terrain (Ishiyama et al. [1991]; Bennett [1997]; Guarnaccia and Quartieri [2012]). However, a 2D representation of the road network can only describe lateral positions, a factor that may limit its use in applications where elevation is important. For instance, consider two buildings at the side of a road, a tall one *behind* a shorter one. A 2D semantic road model would tell us that the entirety of the taller building receives a decreased noise load, because the shorter one – positioned between the taller building and the road – suppresses it. In reality, this observation is invalid because the part of the taller building which is visible from the road above the shorter building receives the full noise load. To be able to represent such 3D relationships in our model correctly, we need both buildings and roads to be truly three-dimensional.

2D-projected digital road models with mediocre accuracy have attracted great scientific and commercial attention since the advent of digital cartography and satellite navigation (Taylor et al. [2001]; Fouque and Bonnifait [2008]; Yue et al. [2008]; Chen and Hsu [2020]). However, *accurate 3D representations* are still atypical, owing to factors such as increased cost of generation and maintenance, increased complexity of visualisation and analysis, and a lack of significant use cases (Zhu and Li [2007]; Wang et al. [2014]). As a result, 2D road models are common in terms of both public and private geospatial providers, whereas accurate 3D road models are rare in comparison.

When a use case arises and an accurate 3D model is needed, the provider generally faces two choices: to produce a new model, or to enrich an existing 2D model with elevation data. The decision generally depends on the quality of the available 2D data set relative to the requirements for the 3D model, as well as that of the dataset(s) available as sources of elevation data, with which the 2D model can be enriched, among other factors (Zhu and Li [2007, 2008]; Wang et al. [2014]). In terms of the source of elevation data, the rule of thumb in the geospatial field is that data acquisition is far more expensive than re-using existing datasets, especially openly available ones. As a result, many providers first attempt to find a way to convert their datasets into 3D using existing data in such a cost-effective manner.

In certain projects the accuracy requirement and restrictions on the modelling procedure may be prescribed legally. Such is the case for the client of the proposed MSc dissertation research, the *Nationaal Dataportaal Wegverkeer* (NDW, National Road Traffic Data Portal), part of *Rijkswaterstaat* (RWS, Directorate-General for Public Works and Water Management), a Dutch government organisation who are in the process of enriching their pre-existing open data 2D road model, called *Nationaal Wegenbestand* (NWB, National Road Database), with 3D data, to attain compliance with *SWUNG2*, the new version of the Dutch noise legislation or *geluidwetgeving*. The new version of this legislation prescribes, among other things, accuracy requirements for the road model underlying the noise simulations, with explicit mention of it having to be three-dimensional. Due to cost considerations and reasons related to NDW's data acquisition pipeline, the pre-existing 2D realisation of NWB will be converted into a 3D dataset (dubbed *3D-NWB*) primarily using open data geospatial datasets. They have produced a prototype realisation themselves, and subsequently contracted the consultant firm *Royal HaskoningDHV* (RHDHV) to create a commercial implementation based on their experience with the prototype. The development of this tool was concluded in December 2020. In addition to simply using the commercial implementation, they wish to assess how it fares in terms of spatial accuracy-related *SWUNG2*-compliance. This dissertation research will attempt to contribute to this assessment by producing an original implementation that favours accuracy, and in which accuracy can be *tracked* and *quantitatively evaluated*. This implementation will also serve as a reference to which its commercial counterpart can be compared so that its accuracy can be evaluated indirectly, and will also explore various related geomatics topics in the process. This document presents the preliminary findings and project plans of the proposed research.

I will primarily focus on a Lidar point cloud and a 3D topographical line dataset as elevation sources. In both of these datasets, it is evidenced that roads are in truly three-dimensional relationships with one another and with their environment. For instance, they cross above and below other roads and are often constructed underground, in tunnels. The question arises, how such real-world geometries should be dealt with in the conversion process. The answer to this question effectively defines which exact field of geoscience my project is positioned in. The two candidates in the context of digital road network modelling are GIS and geomatics. It is thus worth discussing briefly how each typically treats 3D objects. One of the reasons why 2D road models are popular is that their geometry and network properties can be analysed using a multitude of well-proven GIS methods and software kits. However, in GIS models, even if elevation measurements exist, they are generally only present as an *elevation attribute* (i.e. a semantic data field, like street names), because GIS geometrical models do not typically support true 3D operations. This is conceptually identical to projecting the geometries onto the horizontal plane. Geometric models that treat the vertical dimension explicitly are more common in geomatics; namely 2.5D and 3D models. While using 2.5D models restricts the types of physical entities that can be modelled, it also greatly simplifies certain types of analysis conceptually and computationally. This makes it ideal for modelling on similar scales to GIS; on the national scale for instance, as in this research. While 2.5D modelling initially appears to be a good candidate for this project, we may observe that it is by definition unsuitable for handling the 3D relationships that roads have with each other and with their environment. However, much like how the concept of divide-and-conquer works in computer science, it is possible decompose 3D problems into smaller sub-problems until they become natively compatible with 2.5D methods. This research is positioned in the geomatics field because my implementation will explore how 2.5D methods can be applied in a way that enables the *piecewise* modelling of a national road network. The divide-and-conquer concept will be applied to decompose the road network into segments that can be individually, locally regarded as *terrain* and hence be modelled in 2.5D. As the proposed research focuses on 2.5D methods to a great extent, it overlaps with the geomatics field of digital terrain modelling in terms of how it will generate and store the digital representation of road surfaces, and as a consequence, the manner in which it will derive elevations from them. As the Methodology section (5) will reveal, it also strongly overlaps with the geomatics field of 3D feature extraction (and to a small extent, photogrammetry), because of how I propose to derive the 2.5D road surface models from our input datasets. As also indicated in the Research questions section (3), it is among the specific aims of this project to study how a combination of mainly geomatics-based tools and methods can be used to accomplish the tasks required by NDW, as well as to assess their accuracy and suitability when used in this way.

Care will be taken to examine how the same challenges as we expect to face regarding output accuracy and 3D road interactions are handled by the commercial implementation – in fact, some of my research questions are based specifically on challenges reported by the developers of the commercial implementation, and the known shortcomings of their final product. Our preliminary analysis of the implementation indi-

cates that for instance, it uses input data with non-validated accuracy for motorways, and simple bilinear interpolation in poorly interpolated, Lidar-based raster DTMs for provincial roads. Furthermore, it only handles 3D interactions explicitly for motorways and even for these, it does so via a series of assumptions that do not hold in general, necessitating the use of various fallback mechanisms. This comparison may provide an insight into some aspects of 3D geospatial data processing that are frequently overlooked, or oversimplified in commercial implementations.

To summarise, the planned implementation of the proposed research is comprised of taking NWB, applying a pipeline of mostly geomatics operations, and outputting an elevation-enriched version of it. In intermediate steps, it will be decomposed into subsets that are solvable by 2.5D methods, and reassembled into a 3D-enriched version of the input geometries. A byproduct of the procedure will be accurate piecewise 3D road surface models. The accuracy will be tracked throughout the procedure. Ultimately, the implementation will be a *tool* to help answer the research questions that are presented in the next section.

2 Related work

Many relevant research papers have been located and examined as part of the literature review stage of this project. Because my focus will be restricted to openly available Dutch geospatial datasets, I already know which sources of elevation measurements will be relevant. For reasons to do with the selection of available datasets and their accuracy-related characteristics (further discussed in Section 4 and Section 5), my primary candidate is the national Lidar point cloud of The Netherlands, AHN. As a result, there are two main areas that are of interest in this literature review: Lidar accuracy (and derived DEM accuracy), and road feature reconstruction from Lidar point clouds. The literature review concerning these two areas will be presented below in separate subsections.

2.1 Lidar accuracy

Accuracy description of Lidar sensing

Many papers describe that the accuracy of Lidar-derived DEMs depends on the accuracy of the sensing method itself. The notable paper [Hodgson and Bresnahan \[2004\]](#) describes the most fundamental sensing errors of Lidar measurements to be introduced by Global Navigation Satellite System (GNSS, such as GPS) errors, Inertial Navigation Unit (INS) errors, Inertial Measurement Unit (IMU) errors, errors introduced by the waveform analysis algorithm and lastly, a general error factor that depends on the flying height. Combined, these are the primary factors that contribute to the measurement accuracy of a Lidar survey, together making up the nominal accuracy of a given survey, meaning the figures that can be found in the documentation accompanying Lidar data. It is shown by various papers including [Hodgson and Bresnahan \[2004\]](#); [Su and Bork \[2006\]](#); [Kraus et al. \[2006\]](#); [Raber et al. \[2007\]](#); [Peng and Shih \[2006\]](#); [Chow and Hodgson \[2009\]](#); [Aguilar et al. \[2005, 2010\]](#); [Guo et al. \[2010\]](#) that there are local factors influencing Lidar accuracy that are independent of the sensing equipment and are not generally reported in detail by data providers because they are difficult to estimate and vary spatially. This is not to say that are ignored by official figures entirely, for instance horizontal and vertical accuracy are reported separately by vendors because unlike horizontal accuracy, vertical accuracy also depends on the local terrain slope, as shown in Figure 2.1. It is widely regarded that the elevation

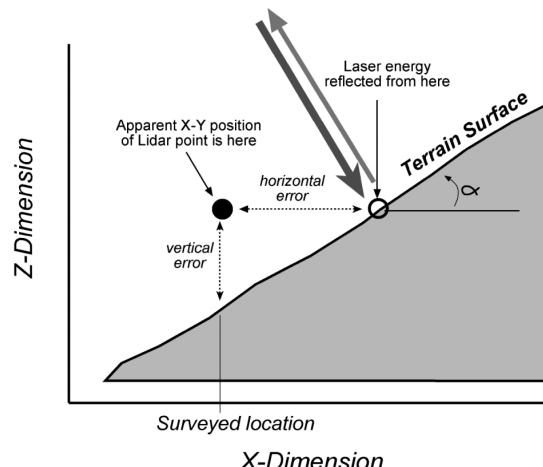


Figure 2.1: Sketch diagram showing the effects of horizontal error on vertical accuracy in the context of sloped surfaces. Any uncertainty in the lateral position of the point of reflection will be scaled by the tangent of the slope angle, denoted by α ([Hodgson and Bresnahan \[2004\]](#)).

error increases linearly as a function of increasing topographic complexity (commonly represented as a 2D slope map), and logarithmically as a function of decreasing local point density (falling off rapidly beyond a certain threshold density). Furthermore, an equivocal consensus also exists regarding the influence of vegetation. In all cases it decreases accuracy, with the significance of the error depending strongly on the type of vegetation. Mature trees and evergreens tend to influence accuracy to a lesser extent, whereas bushes, shrubs and undergrowth in general tend to have a decidedly larger impact. [Peng and Shih \[2006\]](#) quantified this as a function of *vegetation angle*, a qualitative measure (not a real angle) that describes how well Lidar can theoretically penetrate various types of vegetation. They found that there is a linear correlation between elevation errors and vegetation angle, as well as canopy volume. The one exception is [Raber et al. \[2007\]](#) which reported specifically that in very strongly vegetated areas, no correlation could be found between vegetation classes and accuracy (or even point density and accuracy). Research tackling these topics uses empirical methods to estimate errors, which generally consist of surveying ground control points accurately and either directly comparing with nearby Lidar points, or first constructing a spatially continuous DEM and comparing the interpolated values in the DEM with the surveyed reference elevations. The papers also establish that correlations exist between the examined sources of error, most importantly between vegetation and point density, with the latter intuitively decreasing in places of significant vegetation cover. Other correlations have also been reported, for instance between point spacing and vegetation angle, as well as point spacing and slope in [Peng and Shih \[2006\]](#), and a weak logarithmic correlation between point density and slope by [Chow and Hodgson \[2009\]](#).

The point is made in several of these papers that because of the logarithmic correlation between point spacing and accuracy, increasing the target point density of a survey is only justified up to a certain point. This depends strongly on the study area because point density itself is correlated with the vegetation cover and often the terrain relief. In most of the mentioned papers, the pinpointing of specific sources of error and the type of correlation include traditional methods of manual or automatic 1D or 2D regression, as well as for instance supervised classification with potential sources of errors as the variables. It is argued by several authors, most prominently by [Guo et al. \[2010\]](#), that in vegetation-free areas of low relief, most ALS surveys oversample the terrain by as much as 30 to 50 percent, leading to increased processing times, reduced algorithmic stability, and no improvement in accuracy. [Bater and Coops \[2009\]](#) comes to the same conclusion, and the logarithmic trend generally observed between point density and elevation accuracy further supports this. Conversely, in rugged, vegetation-covered terrain additional cross-flight surveys can increase accuracy significantly by improving ground point density, as [Peng and Shih \[2006\]](#) noted.

Accuracy description of Lidar-based DEM-generation

The topic of the influence of DEM interpolators (specifically, DTM interpolators) on accuracy has also been widely studied. There exist various types of approaches, such as deriving exact error propagation formulae from the mathematical descriptions of certain interpolators, as well as the more popular approaches based on simply performing the interpolation and checking its accuracy post-application via split-sample, cross-validation or jack-knife methods. As an example of the theoretical approach, [Aguilar et al. \[2010\]](#) propagates errors mathematically through the IDW interpolator to obtain a specific expression. Among other things, such a formula depends on the sensing accuracy, the local factors of accuracy (e.g. slope), the gridding resolution, as well as a mathematical expression derived from the interpolator's formal definition. As that paper shows, it is possible to simplify the process by performing Monte Carlo simulations on the mathematical definition of the interpolator rather than to derive the formula directly. [Kraus et al. \[2006\]](#) also apply a similar, theoretical method to analyse errors propagating through the moving Maximum Likelihood Estimator (MLE). Post-application statistical evaluation was performed by for instance [Peng and Shih \[2006\]](#) (jack-knife, using surveyed reference points), and [Guo et al. \[2010\]](#) (ten-fold cross-validation). Notably, [Smith et al. \[2005\]](#) used all three approaches (split-sample, cross-validation, and jack-knife) for a wide range of interpolators in an urban setting. Firstly, many of these papers examined the influence of gridded DTM resolution on accuracy. [Chow and Hodgson \[2009\]](#) examined via regression techniques (on IDW interpolation) how it is correlated with point density and found that linear to logarithmic correlations exist. [Guo et al. \[2010\]](#) argues that for most interpolators, the overall trend is linear between accuracy and resolution, up to the scale of the Lidar point density. They also found that differences in accuracy between interpolators were most prominent at the finest resolution. [Bater and Coops \[2009\]](#) found that the local influence on accuracy of slope and point density are mostly invariant relative to DTM resolution. In terms of the accuracy ranking of interpolators, there is a clear consensus that no such ranking exists

that is independent of the size and type of the study area, and the purpose of the interpolation. For instance, the accuracy of piecewise spline-based, quintic-type, kriging and ANUDEM methods were found lacking in the context of their insensitivity to small, sudden changes (such as natural faults in the terrain and anthropogenic modifications thereof) while they were proven to work well for large-scale terrain, as described by [Bater and Coops \[2009\]](#) and [Guo et al. \[2010\]](#) for instance. All reviewed papers agreed that the accuracy of all interpolators decreases the most in areas of high relief and reduced point density, with spline-based, IDW methods generally producing the worst results in such areas, especially for large-scale terrain. Interestingly, the relative importance of interpolation-introduced errors is reportedly low relative to instrument-related errors and surface-related local sources of error, according to research such as [Hodgson and Bresnahan \[2004\]](#) and [Aguilar et al. \[2010\]](#). The former goes as far as to state that the decrease in accuracy after the application of an interpolator is insignificant, or that interpolation may even increase the overall accuracy, although this was observed in densely vegetated areas where point spacing and accuracy are severely afflicted. TIN-based interpolation methods were recommended specifically by [Bater and Coops \[2009\]](#) for complex geometries and found it in their research to be the most conservative in terms of RMSE analysis. [Hodgson and Bresnahan \[2004\]](#) also used TIN-based interpolation in their research, and detected no significant decrease in accuracy following interpolation. Furthermore, [Peng and Shih \[2006\]](#) used TIN-based interpolation in their research, in which they found local influences on elevation accuracy highly predictable. Unlike most papers, [Aguilar et al. \[2010\]](#) considers the accuracy of ground filtering explicitly, and states that its success is a precondition of accurate terrain interpolation wherever the terrain is occluded or shaded partially. It suggests the estimation of ground filtering error (even if only from generic values) and its inclusion in overall elevation accuracy. Overall, it is evident from the review (and specifically recommended by various authors) that testing several candidate interpolators before making a final choice is recommended.

2.2 Road identification in point clouds

Research using point clouds only

Our second domain of interest is feature extraction because our intention is to not only systematically query a DEM for elevations around the NWB road centrelines, but to take into account the entire road surface as represented by the point cloud. This requires one to identify, or at least approximate which point cloud points were reflected from the surface of the given road.

In terms of point cloud feature extraction techniques relevant to roads, I have looked at a range of papers that give account of a spectrum of divergent methods. One strategy, most prominently represented by [Hu \[2003\]](#); [Hu et al. \[2004\]](#); [Zhu and Mordohai \[2009\]](#); [Zhu and Hyppä \[2014\]](#); [Lin et al. \[2015\]](#), is based on the idea of transitioning to a photogrammetric analysis at some point in the process. First, a set of pre-processing techniques to better characterise potential road points in the source Lidar data are applied, generally by performing some form of filtering (in some cases simply by setting thresholds applicable to returns from bitumen), or by extracting ground planes using various techniques and selecting points that lie close to them. Then, images are rendered from the point cloud from various angles, often using colour-coding based on point properties, and applying photogrammetric methods to identify roads. Sometimes high-definition aerial or satellite imagery is incorporated in the photogrammetric workflows. The success rates of such strategies are mediocre, rely strongly on manual parametrisation, and are unsuitable for large study areas with inhomogeneous types and distribution of roads, as also concluded by the literature review in the paper [Yang et al. \[2013\]](#).

A further popular set of strategies rely on road curb detection. [Vosselman and Liang \[2009\]](#), [Zhang \[2010\]](#), and [Yang et al. \[2013\]](#) are examples of such research. [Vosselman and Liang \[2009\]](#) presents a method in which a DTM is generated, points close to it are selected in the point cloud, and small, curb-like jumps in elevation are algorithmically detected. The curb points are selected, and a feature extraction method (RANSAC in this case) is used to construct 3D lines from them. Gaps in the lines are closed algorithmically, and B-splines are fitted to optimise the shapes of the road edges. In [Zhang \[2010\]](#), road cross-sections are inspected in 1D, and the points representing the road surface, the curbs and non-road surfaces are identified. In [Yang et al. \[2013\]](#) an approach is presented in which first cross-sections are identified, ground points are filtered in 1D, and curbs are then identified in the 1D series of ground points.

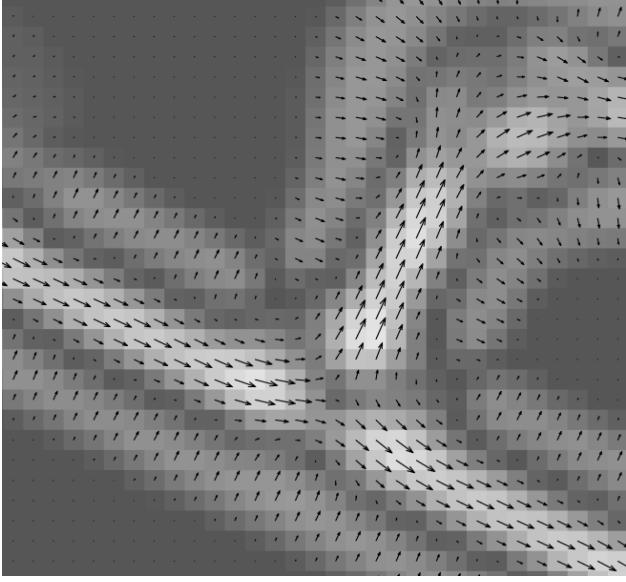


Figure 2.2: Illustration of the results of PCD convolution. The vector magnitude in any given pixel quantifies the support that exists there for the presence of a road centreline, so it is effectively a centreline detector ([Clode et al. \[2007\]](#)).

Their approach is unique because it uses moving window (i.e. convolution) methods on the 1D sections to extract the curbs, testing a subset of the points in each iteration to see whether they satisfy a range of rules. While the results of these methods are more flexible, more accurate and more complete in general than the photogrammetric methods, they are not, in their original form, well suited for this project. The cross-section based approach was originally developed for MLS data, where the data is either natively output in the form of road cross-sections (car-mounted front facing sensor), or can be easily extracted from the point cloud in such a form. [Vosselman and Liang \[2009\]](#) is an exception, which demonstrates that a relatively simple approach can be used to detect curbs without the need for native cross-sections in the data. However, a deeper issue undermines our confidence in the applicability of curb-detection methods. Working with a national dataset and focusing on large roads (including motorways) means that the assumption that well-defined, relatively uniform road curbs will exist and be reliably detectable everywhere is not a sensible assumption.

There are many more papers dealing with this task without using external vector data. For instance, [Clode et al. \[2004\]](#) and [Clode et al. \[2007\]](#) present a set of methods in which first a DEM is generated, then points close to the DTM are selected, are further filtered based on intensity thresholds applicable to bitumen in a hierarchical system, and then the results are refined via morphological operations on the selected regions. The output is a point cloud in which road points are marked semantically as belonging to a road. In the 2007 paper, they extended the procedure by convolving the results with a Phase Coded Disk (PCD), which can create a spatial map of the predicted road parameters wherever it moves through road points. Convolving the PCD is in fact a photogrammetric method, because it acts on a raster generated from the classified point cloud prior to its application. Nevertheless, the method is still more relevant to us than the rest of the photogrammetry-based research mentioned here. An example visualisation of its output is shown in Figure 2.2. They describe it as an alternative to using the Hough-transform for finding road centrelines, which, according to their research, is not reliable enough in this context. This spatial map can then be used to generate a vector dataset describing the geometry of the roads. [Gross and Thoennesen \[2006\]](#) shows that point neighbourhood information can be used to generate covariance matrices of individual points, which can in turn be used to decide directly whether the point belongs to a linear feature. They also describe how lines can be assembled from the selected points efficiently. Other methods relying only on the points themselves exist, but they are typical of real-time MLS applications, such as the fully convolutional neural network-based solution in [Caltagirone et al. \[2017\]](#). The literature review in [Yang et al. \[2013\]](#) offers an excellent overview of such additional methods, but they will not be described here any further, as they are not relevant enough to this project.

Methods using input point clouds and external vector data

The last category contains research that used similar input data to mine (including initial road estimate vector datasets) and achieved similar goals. [Cai and Rasdorf \[2008\]](#) show that enriching road centrelines with elevations can be achieved using simplistic methods. Their first method is based on finding points on opposite sides of roads (in 2D) at similar distances from them, and in suitable locations to form approximate cross-sections. Roads are intersected with the cross-section and the intersections are given elevations by 1D-interpolating inside the cross-sections, using the elevations of the two points defining them. Their second method is even simpler; for each road vertex (or some sampling along its length), they locate the closest Lidar point and associate its elevation with it. The simplicity of these methods is reminiscent of the commercial solution which were developed for NDW by RHDHV (described in Section 5), and highlights that

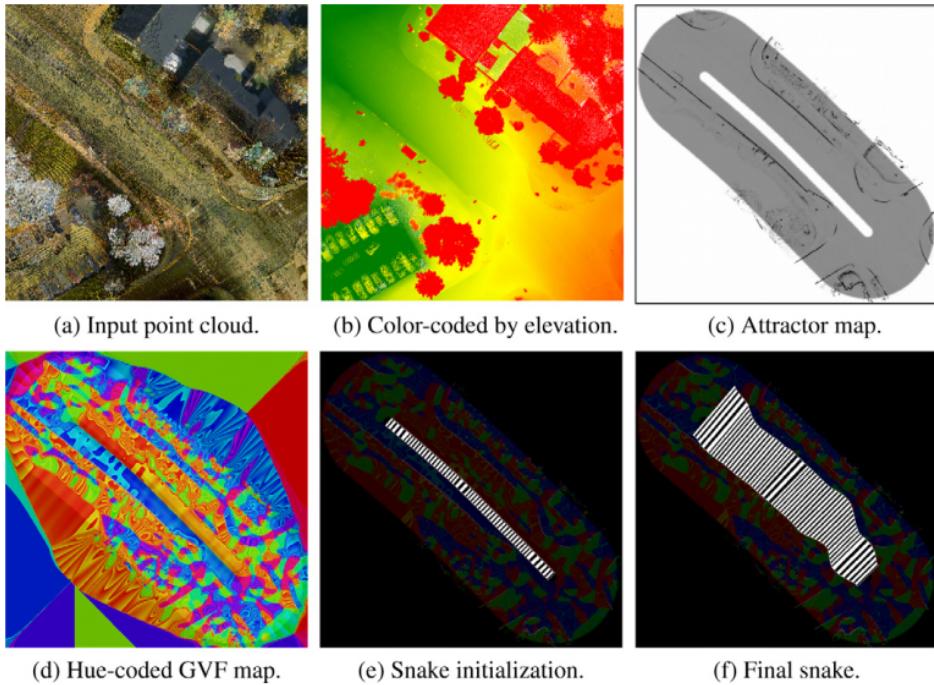


Figure 2.3: Illustration of the procedure used in [Boyko and Funkhouser \[2011\]](#), taken from their paper. In the attractor map, darker colours represent stronger attraction, resulting in the active contour converging to them in the optimisation procedure. The GVF map is a vector map that contributes to favourable active contour convergence and is not described any further here. The initial contour is based on the road centreline (from external vector data).

a rough approximation for the road elevations can, in practice, be made either directly from the point cloud or from a derived DTM in a straightforward manner. However, far more sophisticated methods have been developed by other authors. One landmark research, [Boyko and Funkhouser \[2011\]](#), describes a method in which their input polylines are used to label road points in an ALS point cloud. They first associate the input lines with elevations by fitting spline curves through nearby Lidar points. Suitable Lidar points are selected based on minimising an error function that includes terms related to the distance from the location of interpolation, and to elevation variance. The resulting network is guaranteed to be continuous and smooth because the densified input polylines are used as a spline control points. They then partition the point cloud on disk, based on fitting small support planes along the 3D-converted lines and fetching points that are close to them (solving both the memory and the 2.5D violations related to overlapping roads). They then construct a map penalising points away from road edges based on one term depending on the distance from the road centreline, and another based on a curb detection algorithm (these are so-called *attractor maps*). Lastly, they apply an active contour (snake) optimisation technique that yields the road edges in 3D, and then label points between the two edges as road points. An illustration is shown in Figure 2.3. [Göpfert et al. \[2011\]](#) demonstrates that such active contours can be used to estimate road outlines without the need for the involved pre-processing steps of [Boyko and Funkhouser \[2011\]](#), simply by taking elevations of the input polylines from a derived DTM and optimising the road centreline the same way as its outline (using the active contours). [Hatger and Brenner \[2003\]](#) presents two approaches based on region-growing. The first one is based on growing planes in the entire study area from Lidar points. This being too complex computationally, they propose another approach of treating Lidar scan lines individually, partitioning each into parametrised line segments via linear regression and then inspecting the succession of scan lines and identifying neighbouring segments that are roughly parallel. The resulting groups of (roughly parallel) lines are then treated as planar regions, and an additional region growing step is performed to find any points that might have been left out. The results of this can then either be prepared as a full, 3D-polygon-based planar partition of the study area onto which road polylines can simply be projected, or by further refining the planar partition (eliminating small, meaningless planes) via a RANSAC-based workflow. [Oude Elberink and Vosselman \[2006\]](#) is relevant not only because of the methods it applies, but also the datasets. They enrich the best-known Dutch open data national topographical vector dataset (the present-day equivalent of which is BGT) with elevations, and as their source of elevations, they use AHN data.

They do not exclusively consider roads; all polygonal vector objects are extruded to 3D. Like [Häger and Brenner \[2003\]](#), they propose region growing as the foundation of the elevation extraction workflow. They use the Hough transform to find seed points whose neighbourhood suggest a planar structure, fit planes and then grow by checking the point-to-plane distance of new points, labelling points with the identifier of the plane they belong to. The vector data is then overlaid, and for each polygon the plane is selected which is represented by the most labelled points in its interior. These points are re-fitted a plane, and each such plane is used to extrude the corresponding overlaid vector geometry simply by projecting onto its surface. To improve upon the results of this simple extrusion, they suggest the application of algorithmic topological corrections. Furthermore, to model the interior of the extruded polygons in more detail, they recommend the construction of a Constrained Delaunay Triangulation (CDT) for each, first by inserting its edges as constraints, and the by inserting the Lidar points that generated its surface. The CDT can be refined algorithmically to ensure smoothness. The main limitation of their method is that its method to extract elevations (the region growing approach) is not too accurate, and that it cannot handle overlapping objects, such as roads in motorway interchanges. Some typical artefacts are shown in [2.4](#). Their method was later extended in [Oude Elberink and Vosselman \[2009\]](#) to work well in complex multi-level road settings by using point cloud segmentation in a manner resembling what I already described in the context of [Boyko and Funkhouser \[2011\]](#). In his doctoral thesis [Oude Elberink \[2010\]](#) he combined this method with the overall procedure in [Oude Elberink and Vosselman \[2006\]](#) to form one integral whole. Furthermore he extended it with a road extraction quality assessment procedure, which was later perfected and also published separately in [Oude Elberink and Vosselman \[2012\]](#). Their quality and accuracy assessment methodology is comprised of two separate procedures and the comparison of their results: a theoretical (functional and stochastic) evaluation, and an empirical one against reference data. Interestingly, their reference data was DTB (see [Section 4](#)) and in general, showed good agreement with their road extraction results.

While none of the above research provides an all-in-one solution to answering the research questions of this project, they contain procedures and concepts which can be used as building blocks for my methodology. The main area *not* explicitly covered by the examined papers is the use of external 3D vector data (DTB in our case) as a further constraint when extracting roads from point clouds. However, methods to perform this can be derived from the operations in them that relate to using vector geometries as approximate road locations in 2D, and from general geomatics concepts and methods. An additional consideration related to the input road centrelines is that in all research presented above, refining the lateral position of the road is permitted. However, NDW specifically requested that I do not move the NWB lines horizontally. Hence, I plan to focus primarily on enriching NWB with elevations without moving them laterally, with only exploring its lateral refinement to a limited extent.

3 Research questions

My main research question is "*How can we achieve a 3D conversion of the NWB dataset using Dutch open geospatial data and a primarily 2.5D-based surface modelling methodology, while guaranteeing optimal and quantifiable accuracy and completeness?*". As I already mentioned in [Section 2](#), combining this question with an awareness of which exact datasets I will be able to use suggests two sub-topics: identifying a combination of

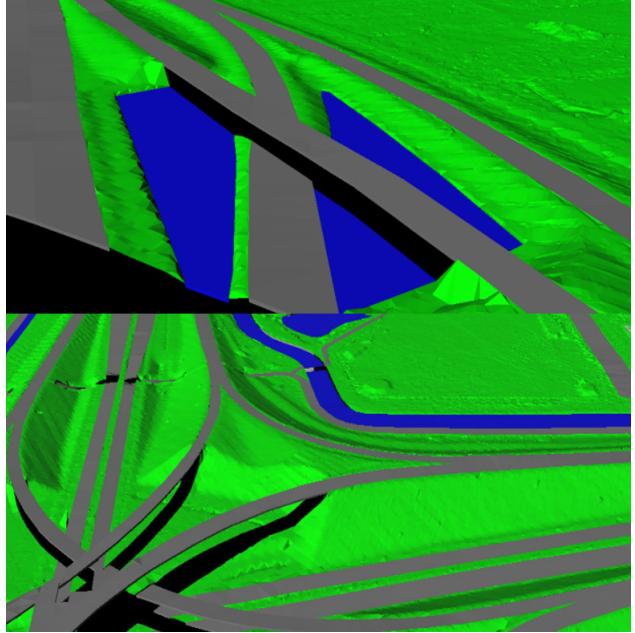


Figure 2.4: Examples of 3D-converted multi-level road vector geometry showing artefacts that are typical in the context of Lidar point cloud to 3D vector geometry conversion. Most notably, in multi-level configurations of roads, only the one with the highest elevation is converted to 3D ([Oude Elberink and Vosselman \[2006\]](#)).

primarily 2.5D geomatics methods that could be used to complete the 3D enrichment of NDW, and ensuring and quantifying output accuracy. The research questions I present below are separated into two groups accordingly.

1. How is it possible to *perform and benchmark* the elevation-enrichment of NWB using Dutch open geospatial data and an efficient, predominantly 2.5D-based geomatics set of methods?
 - a) What are the exact methods of the commercial implementation? What do we suspect the theoretical shortcomings to be, based on RHDHV's methods and output?
 - b) What types of research have been carried out in this particular field? What can we learn from existing research? How successful were the related research projects?
 - c) Can my methods be built around the same datasets as the ones used in the commercial implementation? What are the characteristics of these datasets?
 - d) Is it possible to base the workflow entirely on 2.5D methods by decomposing this intrinsically 3D problem into a collection of smaller problems? Can such a method of decomposition also be used to solve the scaling issues related to handling a national Lidar point cloud input dataset?
 - e) How well does the implementation perform in areas of complex 3D road relationships? How can this performance be assessed visually and quantitatively using real-world examples, such as multi-level motorway exchanges and roads on long, elevated civil engineering structures?
 - f) Is it possible to build the workflow in a way that it allows efficient update operations to be carried out as new data arrives or old data is updated?
 - g) How do temporal inconsistencies manifest themselves in the procedure, and in the output? What problems may arise, and how best to solve them?
 - h) How well does the implementation perform in areas where input elevation data is scarce, or missing? How can this performance best be assessed visually and quantitatively using real-world examples, such as dense vegetation cover or other objects occluding roads, and roads constructed in tunnels?
 - i) How can we best assess the overall performance of the implementation against the commercial implementation in locations that are expected to be difficult to work with, such as where data is scarce?
 - j) How smooth is the output? Are sudden jumps introduced after aggregating the decomposed model? If so, can this be resolved by optimising the procedure, or are additional smoothing steps necessary?
 - k) Can the same workflow be used to also derive elevations for lines that lie a fixed distance away from the NWB centrelines, representing the *vicinity* of roads?
 - l) Can the workflow be used to optimise the *horizontal* location of NWB centrelines?
 - m) Can the workflow serve as an aggregator of elevation data originating from small scale sources such as road management datasets?
 - n) The workflow is planned to produce surface models of *road segments*. What would be needed to aggregate these into a global model containing all roads?
2. Can the processing workflow be developed in a way that it can guarantee optimal *completeness and accuracy*, based on a scientifically sound quantification thereof?
 - a) What types of accuracy may we distinguish between? Topological accuracy? Lateral accuracy? Purely elevation accuracy? How does each type of uncertainty affect the output? Is uncertainty also influenced by geographically variable factors?
 - b) In pre-existing research, what methods have been used to measure accuracy and estimate the influence of processing steps on accuracy?
 - c) What is the accuracy of our elevation sources? Can we trust elevation data sources with undocumented accuracy?

- d) What is the effect of uncertainty in the lateral positions of NWB centrelines? Could the impact of this be reduced by optimising the lateral location of NWB?
- e) What do we need to be able to track the evolution of the accuracy throughout the workflow? Can we build it exclusively from steps that prevent serious degradation to the accuracy, while making any degradation quantifiable?
- f) How does the addition of small-scale elevation data sources influence accuracy? How can they be best made use of in this context?
- g) Would it be possible to develop the solution in a way that it can automatically indicate where levels of uncertainty drop below the prescribed threshold, and why?
- h) If smoothing steps are necessary as a post-processing step, how can we be certain that the vertical displacements do not corrupt accuracy? Are topological constraints necessary? If so, how can they be enforced?
- i) How may we measure or estimate the accuracy of the commercial implementation? Can it be tracked throughout their procedure starting from input accuracy, or is it necessary to rely on more indirect methods?

4 Tools and Datasets

In terms of datasets, both the commercial implementation and the proposed research rely on the following three sources:

- Nationaal Wegenbestand (NWB, National Road Database)
- Actueel Hoogtebestand Nederland (AHN, Current Dutch Elevation)
- Digitaal Topografisch Bestand (DTB, Digital Topographic Database)

As mentioned in the section about research questions, I am also planning to examine how specific datasets from individual NDW data suppliers (such as municipalities, road authorities, civil engineering agencies) can be integrated in the procedure and how they affect the accuracy. At this stage in the project, the specific datasets have not been selected yet.

4.1 Nationaal Wegenbestand – NWB

NWB (or more specifically, NWB-Wegen, the NWB roads product) is an *ESRI Shapefile*-based vector dataset comprised of a semantically rich set of 2D *MultiLineString* objects. They are interconnected in such a way that NWB can be regarded topologically as a graph representation of the Dutch road network. Although the NWB contains all named and numbered roads in The Netherlands, we are only interested in roads that are semantically marked as state-owned or province-owned (category R for Rijk and P for Provincie), because the new noise regulation is only relevant to these types. NWB and the road types it contains are illustrated in Figure 4.1. In addition to its important topological graph structure, the NWB lines are georeferenced with good accuracy. Although the accurate nature of its georeferencing is specifically mentioned in the documentation of the data, it is not evidenced by a rigorous evaluation. The quality description includes only a single figure, which is 5-metre accuracy at two standard deviations (i.e. 95% confidence), but with the confidence being based on the total number of roads, implying that the accuracy evaluation method is performed on a road-to-road basis, each of which may be comprised of multiple *MultiLineString* objects. It is not described, how the accuracy is evaluated for each road, for instance whether it is based on the accuracy of the vertex locations of the road, an arbitrary sampling along their length, and how it is aggregated. Furthermore, we do not know whether the reference for the accuracy was empirical (surveyed control points) or whether it is purely theoretical and is based on the sources and the methods involved in creating and updating the dataset. Both the topological and the geographical information content of NWB is assembled from a wide range of providers ranging from large national providers such as RWS and Kadaster, to local providers such as specific road authorities and civil engineering agencies. No clear



Figure 4.1: An example render of NWB. The road segments (*wegvakken*) are colour-coded to indicate management (*wegbeheerdersoort*). Yellow is used for R-roads (state-managed roads), red denotes P-roads (province-managed roads), magenta means G-roads (municipality-managed roads), and W-roads (RWS-managed roads) are not shown in this figure. Blue roads (T-roads) are roads which do not fall into the above 4 categories, in this case they are managed by TU Delft. This render also contains examples of challenging 3D relationships. Where the P-road Kruithuisweg/Kruithuisplein crosses the G-road Schoemakerstraat and the R-road A13, no intersection occurs in 2D because the roads cross over one another in 3D. These locations are indicated in the figure by red circles. Furthermore, where the bike path IJsmeestertunnel (a G-road in NWB) crosses the A13, a series of 3 short tunnels are located in reality. This is indicated in the figure by a blue oval. These are two types of road layouts that we expect to have to dedicate special attention to.

indication is given in the documentation about the sources used for the compilation of specific NWB road types or the estimation of their accuracies, but it is mentioned that they are inconsistent across the range of road types (Rijkswaterstaat [2020c]). Although the above nominal lateral accuracy (and its description) is not sufficient for compliance with the new noise regulations, the 3D conversion of NWB is not concerned with improving it – in fact, it is a requirement of the project not to displace NWB horizontally. Hence, we may say that the intended outcome of the 3D conversion is to devise a method that can produce accurate elevations for NWB assuming it is up to SWUNG2 standards. To achieve this in practice, lateral refinement is being done in a separate project by developing a workflow to match NWB roads with roads in another Dutch national open data geospatial dataset called Basisregistratie Grootchalige Topografie (BGT). Since BGT is already compliant with the accuracy requirements, correcting NWB geometries based on BGT will foreseeably solve the lateral accuracy problem of NWB. However, at the time of writing of this report, the BGT-based correction has only been carried out and released for municipality-owned roads (category G for Gemeente), hence both the commercial and the scientific 3D-NWB project still rely on NWB data that lacks this correction (NDW [2020]). Regardless of improving NWB's lateral accuracy not being within the scope of this project, it is important to be aware of the uncertainty because of its implications when overlaying NWB with other datasets, such as AHN and DTB.

4.2 Actueel Hoogtebestand Nederland – AHN

AHN are airborne Lidar (ALS) survey datasets available as open government geospatial data in The Netherlands. The surveys are commissioned every few years with AHN1 to AHN3 already complete (1996 to 2019), and the first AHN4 surveys nearing release. At this stage of the project, we are interested in AHN3, although it is a planned side-track of the research to examine how freshly released AHN4 data could best be made use of. The dataset has a combined systematic and stochastic elevation error of 15 centimetres at two standard deviations (i.e. 95 % confidence), and 6 to 10 points per square metre point density on

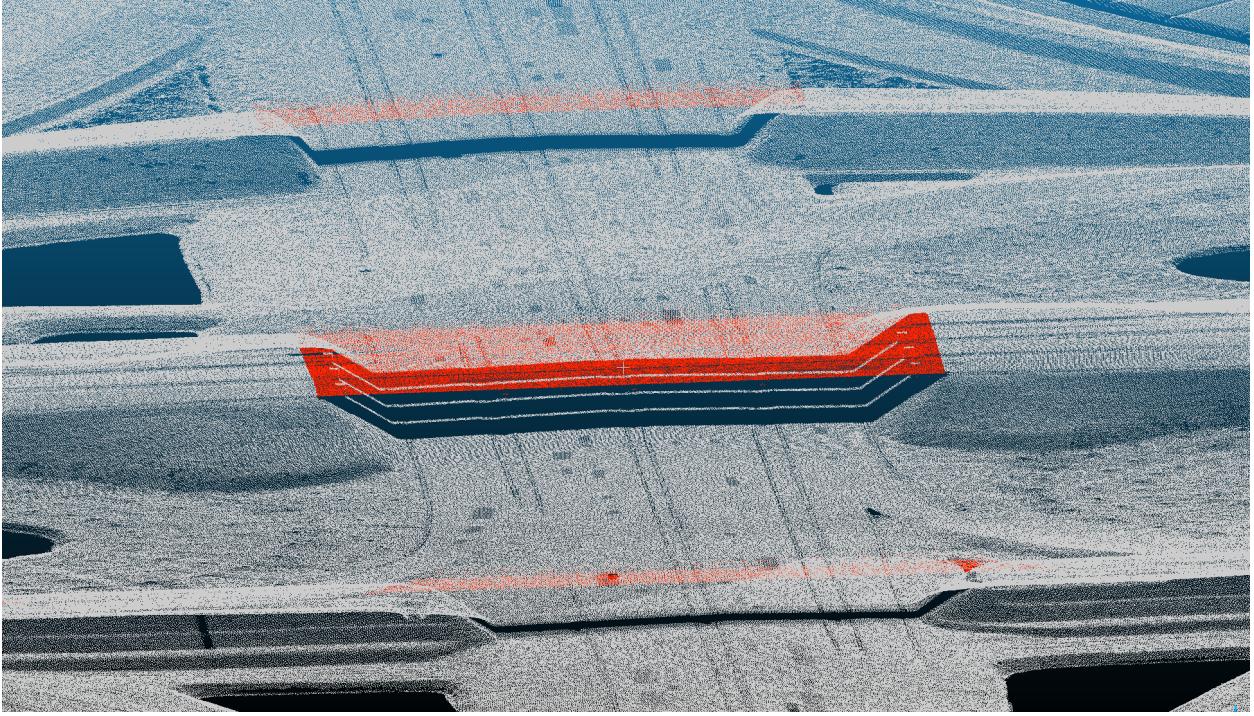


Figure 4.2: An example render of AHN3 with the original point density (no thinning applied) at the Knoppunt Deil, SW of Geldermalsen. The white colour corresponds to ground points (class 2), red corresponds to bridges (class 26). All other classes were removed. The shapes of motorway lanes and ramps, as well as bridges are clearly defined. Road constructed on elevated ground is reliably classified as ground, bridges are cleanly split off by the classification. Even under thin bridges, data becomes gradually sparser at the border and becomes extinct directly below the bridge. This render also makes it clear that wherever vehicles were encountered, the sampling is worse locally. It does not go entirely extinct however, as most locations have been scanned multiple times.

average, corresponding to a 0.41 to 0.32 metre posting distance. The lateral error is 18 centimetres at two standard deviations ([Rijkswaterstaat \[2020a\]](#)). Comparing these to the descriptors of datasets mentioned in research examined as part of our literature review (see the Relevant work section), AHN3 can be considered accurate, and to have excellent point density. The AHN3 point cloud has been classified semi-automatically with great accuracy and is released with the classification included. This means that for most purposes, the point cloud needs not be ground filtered manually, and that extracting certain features is made far easier. Of the various classes available, we are interested in class 2 the most, which corresponds to ground points, and class 26 which contains, among other things, bridges. Ground points are primarily interesting to us because they contain the road points. This includes roads that were constructed directly on the terrain, as well as roads constructed on altered terrain, such as those built on elevated ground or in open trenches. It also contains the points that represent the terrain in the vicinity of the roads. Elevated roads and bridges are not considered ground points and are generally represented by gaps in class 2, which can, in most cases, be filled by using data from class 26, as shown in Figure 4.2. However, it is worth noting that class 26 also contains other types of objects, for instance large motorway signs arching over the road surface (shown in Figure 4.3), as well as the civil engineering structures of elevated roads and bridges (in addition to the road surfaces on them). AHN3 is released in the form of a point cloud, as well as DSM and DTM rasters. Both DEMs were produced using basic radial IDW interpolation with a fixed parametrisation. The DTM was generated by including only points from class 2 in the interpolation step. They are available at 0.5-metre and 5-metre resolutions, with the 0.5-metre resolution being relevant to this project in terms of the target accuracy. For context, this raster converts (on average) 1.5 to 2.5 points into a single raster cell. The Lidar tiles are released in the binary LAZ format (compressed LAS, or LASzip), and are generally several gigabytes in size. As each tile only covers an approximate area of 32 square kilometres and the LAZ compression ratio for this dataset is 0.1 on average, scaling considerations are relevant to this project.

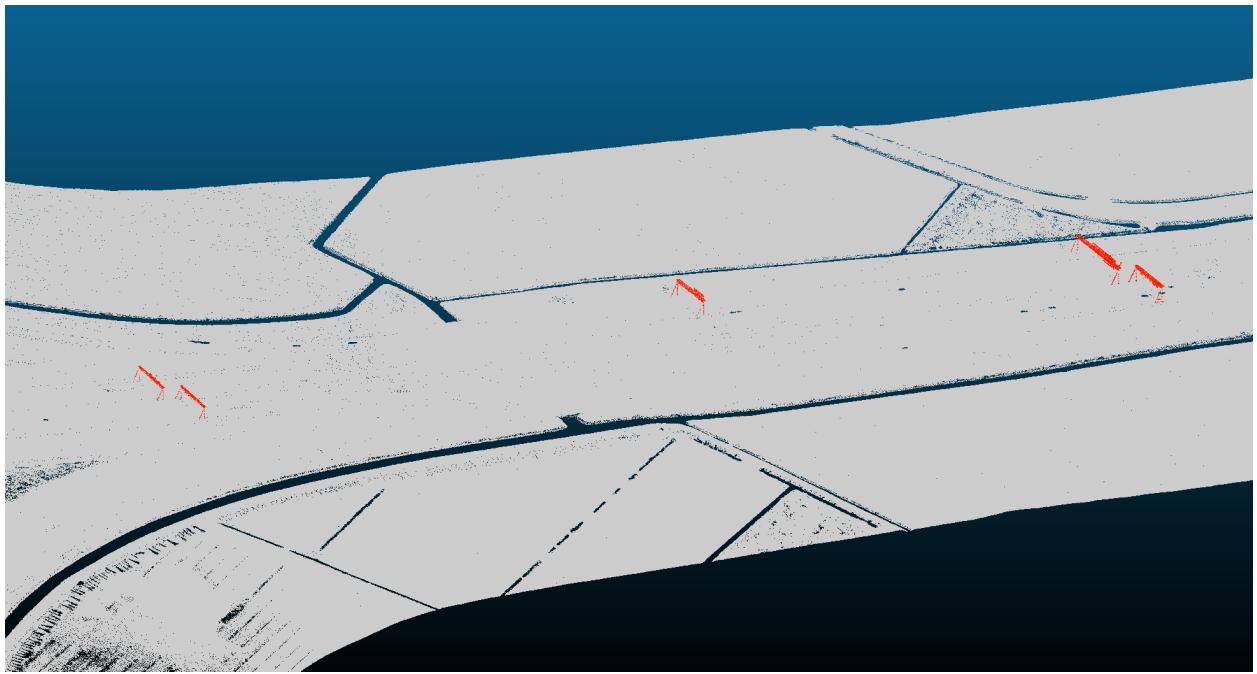


Figure 4.3: This render is from the same location and data as 4.2. Here it is visible that my testing datasets, all points falling outside a 150-metre buffer zone around the NWB centrelines have been removed. While class 26 primarily contains bridges, this image shows that full-width motorway signs are also contained by it. It also shows that the presence of water results in holes (the dark, linear features running parallel with the motorway in this render) in the ground-only point cloud.

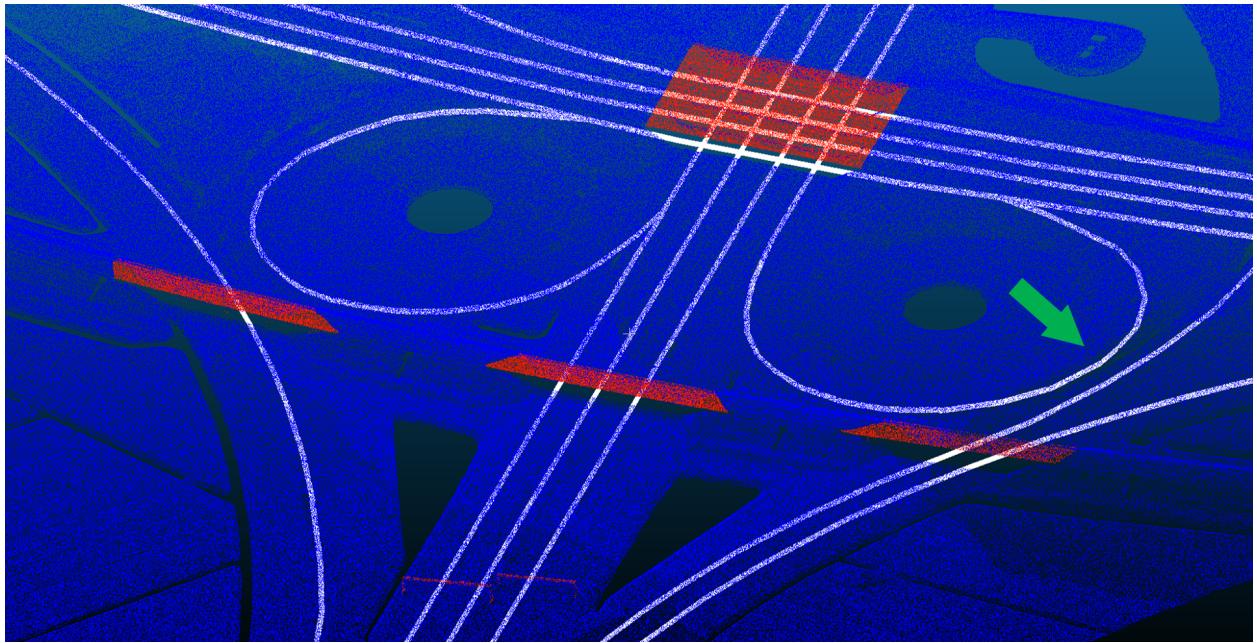


Figure 4.4: This render is from the same location and data as Figures 4.2 and 4.3. Here, ground points are coloured blue, and NWB centrelines are shown in white. As they have no elevation, they appear below AHN3 points and are consequently masked out partially by them. In general, the visual agreement between AHN3 and NWB appears to be good. The most noticeable deviations from the AHN3-defined centrelines of roads occur where they are intensely curved, and where motorway ramps merge into motorway lanes. The reason for the prior appears to mostly be the angularity introduced by the finite discretisation of curves in NWB, for instance in the location pointed out by the green arrow. The succession on three bridges in the bottom of the image do not have an associated NWB centreline because they are railway bridges. The same would be observed for roads not yet registered in NWB (i.e. newly roads).

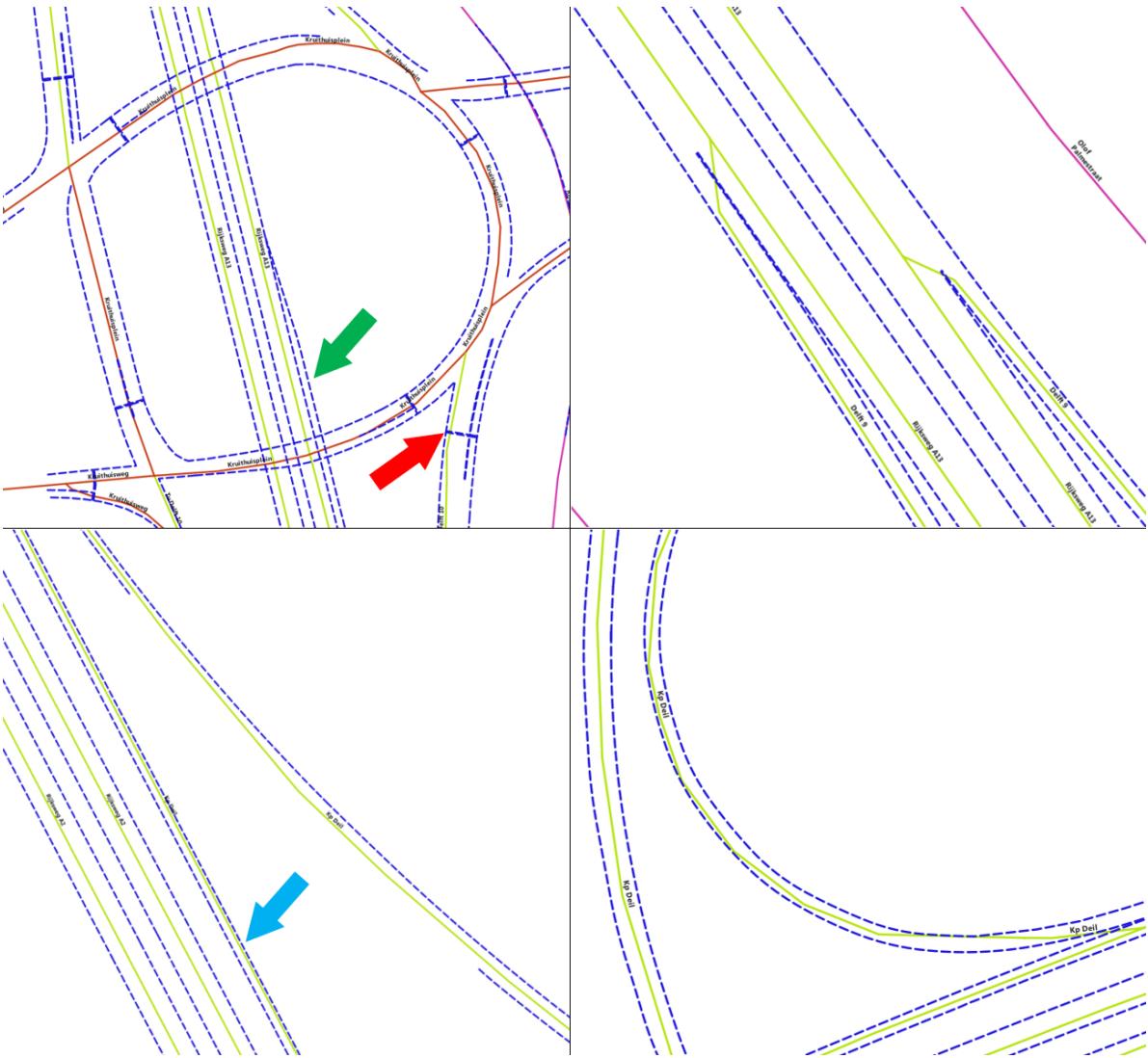


Figure 4.5: Renders of DTB *verflijnen* overlaid with NWB centrelines, illustrating “compatibility issues” and individual DTB issues. NWB symbology is identical to the one used in 4.1, blue dashed lines are the DTB lines. **Top left:** there is not necessarily only one *verflijnen* line on either side of NWB (green arrow). Furthermore, stop lines are also often in this category. **Top right:** NWB motorway ramps often merge with the motorway lanes at unrealistic angles. This causes NWB to intersect DTB lines at these locations. **Bottom left:** DTB road edges are often missing. Furthermore, NWB centrelines may be very close to DTB edges, as indicated by the blue arrow. **Bottom right:** NWB discretisation and inaccuracy often results in angular lines in sharp bends. NWB often gets close to, even intersects DTB in such places. Although this is a 2D visualisation, DTB has 3D geometry as Figure 4.6 illustrates.

4.3 Digitaal Topografisch Bestand – DTB

Like NWB, DTB (or more specifically, DTB-Droog, the DTB roads product) is a Dutch open data geospatial dataset in the *ESRI Shapefile* format. For the purposes of this project, we only need DTB street outlines (edges), hence we can also regard this as a dataset comprised exclusively of *MultiLineString* objects, making it identical to NWB in its data structure. DTB is managed by RWS and it is concerned only with state-owned roads and roads on state-leased land, which translate almost exclusively to NWB R-roads. DTB is, hence, not a reliable source of information about provincial roads (NWB P-roads), our other road type of interest. The DTB lines that are commonly used as road edges are not, in fact, the edges of the paved surface. They are called *verflijnen*, and are the approximate locations of the painted lines representing the outermost edges of the area open to traffic. This is a lucky correspondence with NWB, as it too, contains the centrelines of the areas of roads that are open to traffic. The second-best option would be to use the category *geleiderail constructie*, which generally provide good approximations of the edges of the paved surfaces on

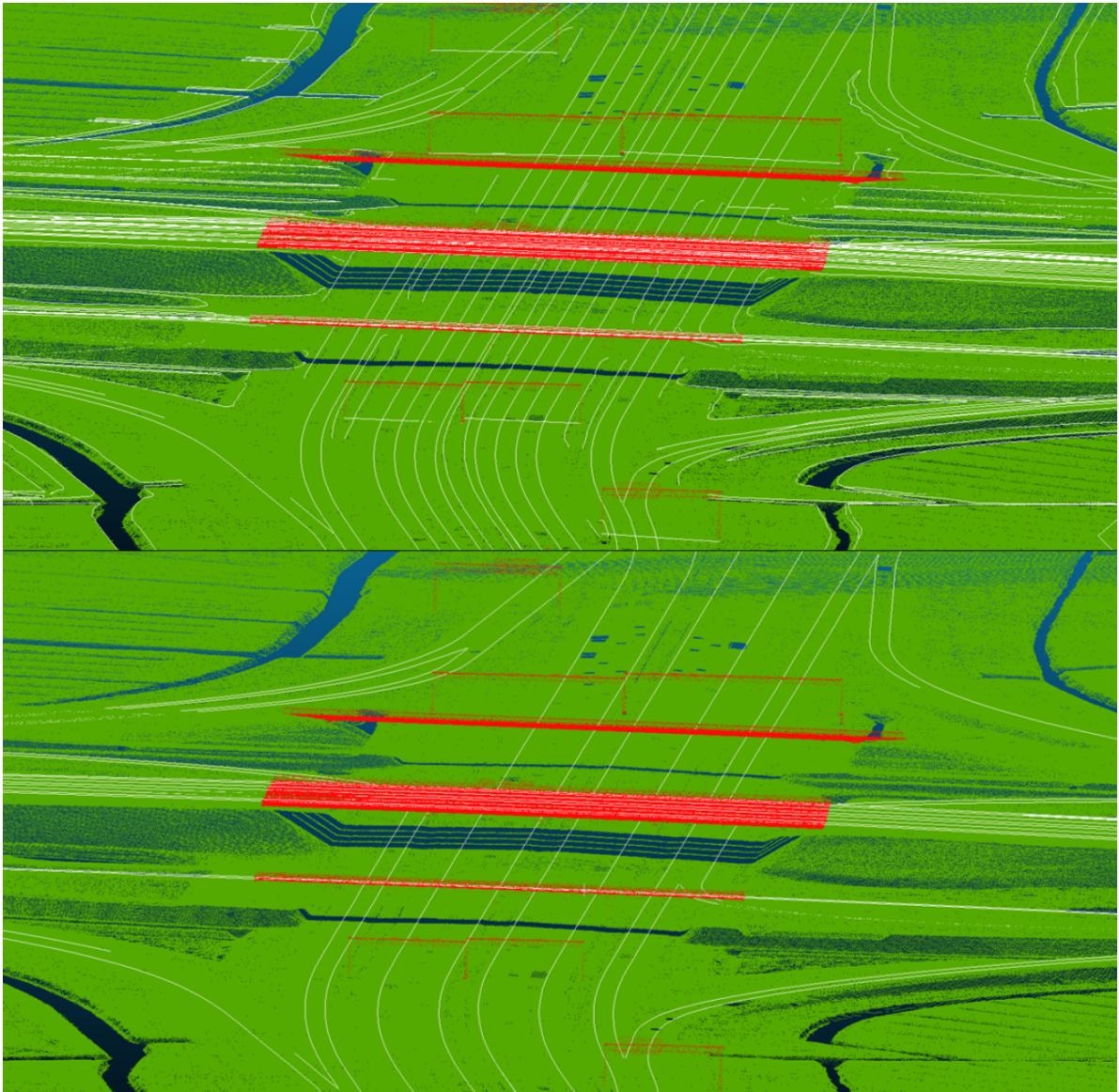


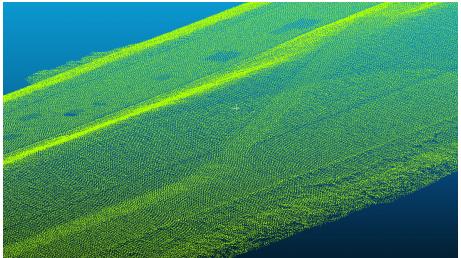
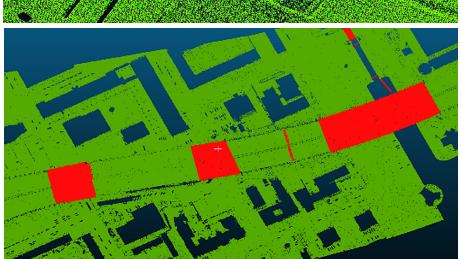
Figure 4.6: Renders of DTB line features overlain with AHN3 in 3D. Ground points are shown in green. **Top:** all DTB lines are shown, illustrating that other features, such as roadside slopes, canals, ditches, safety rails are represented in DTB. Full-width motorway signs are occasionally represented (either by lines on the road, or on the structure itself, inconsistently). **Bottom:** Only *verflijnen* shown. The lines are generally in good agreement with AHN3 visually. This figure also illustrates that they may be useful for identifying which class-26 Lidar points lie on road surfaces, as they contain good first approximations for their elevations.

which the *verflijnen* were painted. Unfortunately, these correspond to the location of safety rails, and as such they are sometimes not on the road and are also not always present. Like NWB (but unlike AHN), DTB's acquisition also concerns several organisations performing various types of sensing, which are then semi-automatically assembled into the complete DTB product. DTB lines are primarily photogrammetry-derived, with land-based surveying methods and car-mounted Lidar (MLS) used in some areas to improve accuracy ([Oude Elberink and Vosselman \[2012\]](#)). The documentation of DTB consist of a documentation ([Rijkswaterstaat \[2020d\]](#)) and a handbook ([Rijkswaterstaat \[2020b\]](#)). The accuracy of DTB is not published by RWS or any other organisation contributing to or using the dataset, although it is generally thought to be quite accurate. It is mostly used for purposes relating to road management and civil engineering and apart from road edges, it contains more than 400 other types of semantically identified roadside object types. As most users of DTB work in GIS environments, particular care is taken to separate objects into layers in such a way that inside any one layer, they never overlap. This means that they can be processed using 2D GIS tools easier, reducing the significance of elevation to a mere semantic detail (non-geometric

attribute). Whenever objects in any given such layer are found in the same position (or closer than a certain threshold), they are automatically shifted by a maximum distance of 5 centimetres to eliminate the overlap, most commonly for DTB road signs. While descriptions in the documentation can be found suggesting that DTB lines are measured accurately, I must consider the formal accuracy of DTB unknown due to the lack of any numerical figures or other evidence to support this. Furthermore, as I illustrate in 4.5, a visual comparison with NWB reveals various types of significant inconsistencies and interpretation issues. A visual comparison with AHN3 is also shown in 4.6.

4.4 Testing data

Due to its volume, it will not be possible to develop and test the implementation without cropping AHN3 input point cloud tiles. Even after the implementation of the segmentation workflow, the size of the testing point clouds will need to be manageable, to facilitate efficient debugging. At the same time, they will need to be representative enough to provide an insight into the computational complexity and visual performance of the algorithm. I have already produced a range of testing datasets by selecting various lengths of NWB centrelines at interesting road features, and keeping only those Lidar points which are within a 150-metre buffer distance from the selected road centrelines. Table 4.1 provides an inventory of these. The tile identifier in the first column refers to those used in PDOK [2020]. The resulting datasets fit into memory easily, even when extracted to LAS.

Title, tile	Features	Render
Markerwarddijk, 20BN1	Dike-based P-road in the Markermeer. Very limited amount of terrain around the road. Road consistently built on ground, with no involvement of bridge structures.	
Amsterdam Hemhavens, 25BZ2	Ringweg-West (R-road) as it crosses the IJ through the Coentunnel in a densely built-up environment. It is built on artificially elevated ground and on bridges in this area. Part of Westradweg also included, which has been built entirely on a long bridge.	
Amsterdam Zuid, 25DN2	R-roads with many closely spaced bridges, small tunnels, dense grouping of holes around roads due to presence of water and buildings	
Bunschoten, 32BN1	P-roads with three big roundabouts, and one road that ends in a small roundabout. Amersfoortseweg has its two lanes on separate roads surfaces (like motorways), but with frequent connecting segments.	

Veluwe, 32FZ2	Straight R-roads surrounded by dense forest and crossed by wide wildlife overpasses.	
Apeldoornseweg, 32HZ2	P-road in dense forest with canopy frequently occluding the road surface, decreasing point density and occasionally creating gaps. The road has small parallel branches running very close to it, which may make it difficult for the algorithm to distinguish between them. It also has roundabouts.	
Hoenderloo, 33CN2	P-road in extremely dense, continuous forest with canopy frequently occluding the road surface, decreasing point density and occasionally creating gaps. Both lanes are on the same road surface	
Rotterdam Ketheltunnel, 37EZ1	This segment of the A4 heading North from Rotterdam has been recently reconstructed in an underground tunnel. In addition, a significant portion of this R-road now runs in a trench towards Delft. AHN3 was imaged during the reconstruction, and hence contains erratic data about the road surface.	
Knoppunt Ridderkerk, 37HN2	The Ridderkerk interchange is one of the largest of its kind in The Netherlands, in one place containing 4 overlapping R-roads. Furthermore, it contains an extremely high density of R-roads in a small area, many of them very tightly packed in small areas. Many are intensely curved.	
Gorinchem, 38GZ1	Complex interchange between a P-road and an R-road with small ramps, roundabouts and overlapping geometries.	
Knoppunt Deil, 39CZ1	Less complex, but analogous interchange to the Knoppunt Ridderkerk.	

Please refer to Figures 4.2, 4.3, 4.4 and 4.6.

Table 4.1: Inventory of testing datasets

4.5 Tools

In terms of tools, for visualisation purposes I will use *QGIS* and *CloudCompare*, with potential additions in later stages of the project. In terms of coding, the source code will be written in *Python 3*, with a strong dependence on pre-existing packages. The final set of tools will be determined when implementing the workflows. To give a few examples, I may use native modules such as *fiona* and *shapely* for handling basic vector operations, and *PDAL* and *CGAL* bindings as well as binaries such as *GDAL* and *LASTools* for point cloud processing and various geometric operations such as terrain modelling. *GitHub* will be used for source code version control.

5 Methodology

This section first presents a top-level description of the methodology of the planned dissertation work. This description is limited to matters relating to planning the execution of the dissertation work. Methodology is then further discussed in terms of the exact *methods* used in the commercial implementation, and the ones that I plan to implement as part of the dissertation work. First, the workflows are presented which were implemented by NDW to produce the prototype implementation and by RHDHV to build the commercial toolbox. This is followed by structured *first approximations* of the workflows I will aim to implement and analyse during the period starting from my P2 presentation and leading up to the P4.

5.1 Top-level methodology

Since this project concerns a client with specific requirements and pre-existing methods in addition to aspects that are purely scientific, the first stage of the project involved *consultation with the client* (and directly with their commercial developers) in addition to the general task of *familiarising myself with the research topic and literature*. The results of these preparation tasks were *discussed internally* (with my supervisors) and used to *define the final list of formal research questions*, and to *produce the P1 submission*. This stage roughly coincided with Q1 of the academic year, and with the P1 period of the dissertation research.

The second stage of the project involved *further consultation with the client and their developers* to determine to what extent the commercial and scientific branches of the 3D-NDW project could be linked, and to allow me to understand the exact methods used in the prototype and the commercial implementation (which was being actively developed during this period of time). In parallel, I *performed the necessary in-depth literature review* and preliminary analysis. The preliminary analysis comprised a *close examination of the input datasets and their documentations* and based on this and the research questions, the *final selection of relevant concepts and methods*. I also selected a range of illustrative geographical regions during the preliminary analysis, and cropped the datasets to their extents to *create testing input files* for later development and testing. Lastly, the results of this stage were distilled to *produce the present P2 proposal*. This stage roughly coincided with Q2 of the academic year, and with the P2 period of the dissertation research.

The third stage of this dissertation will span the period between my P2 and P4 presentations. This stage will be concerned with performing the bulk of the analysis. The period will start with the *implementation of individual algorithms and steps of the workflow*. The first approximation of the *overall workflow* will be refined iteratively, as a function of what I find to be feasible to implement considering the available set of tools, as well as the scope and time constraints of the project. Following the P3 meeting, I plan to shift my focus to *assembling the pipeline* from the individual algorithms and procedures and when finished, *performing the accuracy-related analysis*. Testing the individual procedures and the pipeline will be continuous and be carried out regularly while debugging the code during development, as well as in more depth after reaching milestones. Testing, and the assessment of performance and accuracy will use the testing datasets I produced before the P2 date. This stage will be concluded by *writing the draft thesis* based on the results.

The last stage of the project will last a few weeks between the P4 presentation and the P5 presentation. Both the *implementations and the thesis itself* will be improved and finalised during this time.

The above methodology is illustrated on a flowchart in Figure 5.1. Furthermore, the planned schedule of the tasks is shown on the whole-page diagram in Section 6.

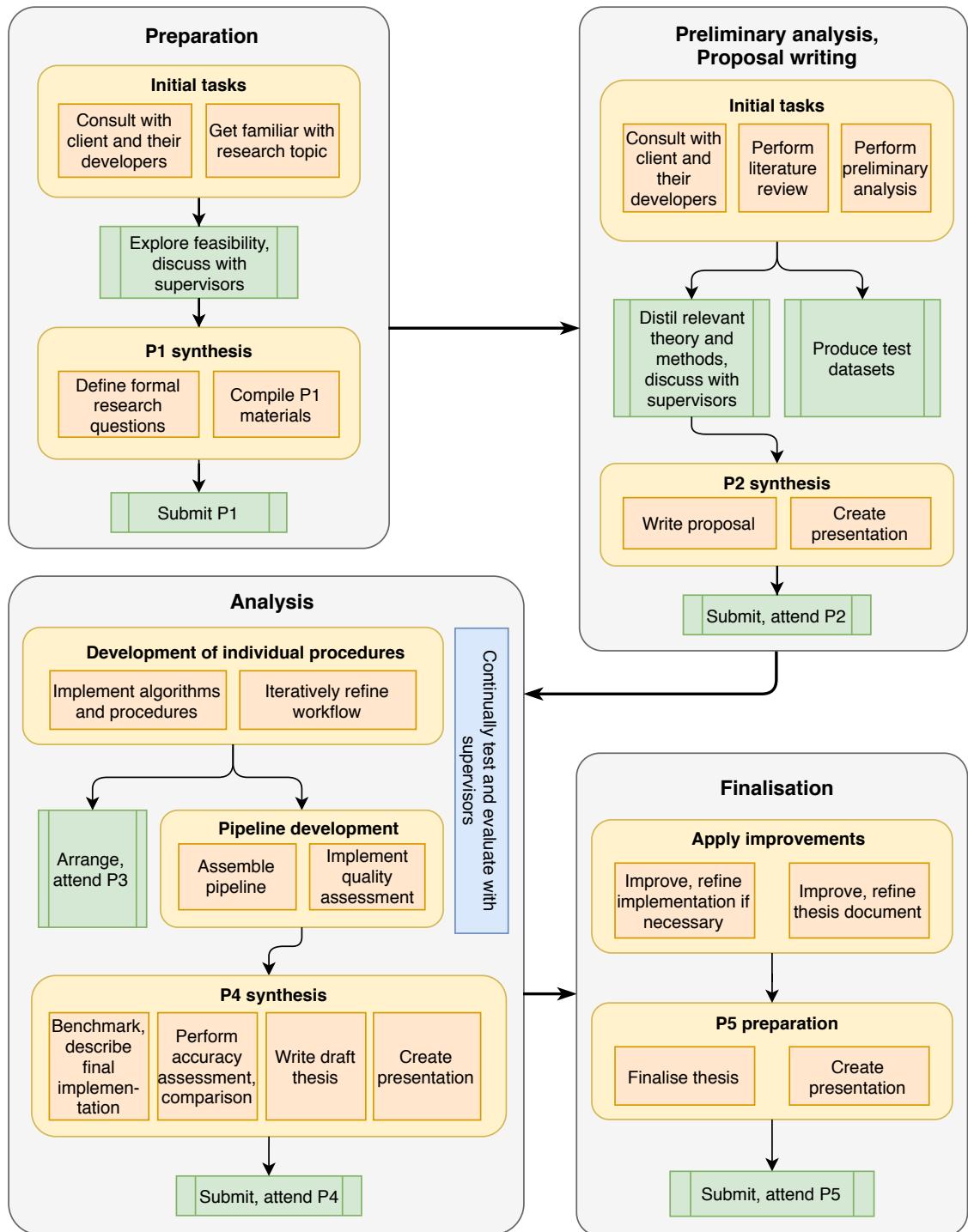


Figure 5.1: Flowchart-style illustration of the top-level methodology of my proposed dissertation research.

5.2 Methods of existing implementations

NDW prototype

NDW themselves produced a non-commercial prototype implementation, which can achieve a 3D enrichment of NWB with only a few gaps in the produced elevation profiles. Although their workflow does not have a formal documentation, I have been given a verbal description of it, as well as the output. Based on my understanding of these, their primary technique involved snapping close-by AHN3 Lidar points to the line geometries of NWB. Notable problems with the implementation included non-road points being snapped to centrelines, causing road centrelines to be given overestimated elevations, in turn resulting in sudden jumps in the elevation profiles. Furthermore, no close-by points could be found for underground roads (i.e. tunnels), and strongly occluded parts of roads. For small gaps, this was resolved partially by writing an algorithm to interpolate linearly inside NWB, using the closest vertices where snapping was successful. For larger gaps, an attempt was made to resolve issues by including information from external sources semi-automatically. Neither issue could be fully resolved via these approaches, hence the results of this project were only used by NDW to gain a better understanding of the problem and the expected challenges. For a reliable, commercial toolbox they subsequently commissioned an implementation from RHDHV.

RHDHV commercial toolbox

RHDHV developed their implementation in parallel with the *planning* of the present scientific research. I have attended NDW-RHDHV meetings, discussed the implementation directly with RHDHV, and was granted access to the codebase of the project. Understanding their implementation is crucial for this research, as it aims to both assess the accuracy of the commercial implementation, as well as in part base its methodology and implementation on the suspected shortcomings of the commercial implementation. The steps of the procedure are briefly described below.

First, where NWB vertices are too sparse, densification takes place; additional temporary vertices are created inside NWB line segments. Then, *different* workflows are initiated for R-roads and P-roads.

For P-roads (for which DTB data does not generally exist), the workflow is conceptually similar to the one in the prototype, with the notable difference of using AHN3 DTM rasters rather than the point cloud. Because AHN3 DTM rasters are badly affected by both large holes and small groups of missing pixels (due to the fixed-parameter IDW interpolation that was used to generate them), RHDHV could only make use of them by filling in the gaps of the raster using linear interpolation in a 3D TIN created from extruded raster pixel centres. They overlaid the raster tiles with NWB vertices (including the dense, temporary vertices) and interpolated their elevations using bilinear interpolation inside the raster.

For R-roads DTB is available, and the assumption is made by RHDHV that it is more accurate than AHN3 (or at least the stock DTM tiles generated from AHN3), to the extent where it should be the primary source of elevation data. Priority is thus always given to it in the procedure, with AHN-based interpolation used only as a fallback mechanism in case DTB-based height estimation fails. The goal of the procedure is to find the DTB line segments that delineate the road edges at any given location in the NWB and deduce elevations from them. First, 2D cross-sections are constructed on NWB vertices, with each given the mean azimuth of the two NWB line segments that they are part of, and also on densified vertices, which receive azimuth values based simply on the azimuth of their parent line segments. DTB lines are then intersected with the cross-sections and for each cross-section, the closest DTB line that satisfies a relative angle condition is picked on both sides. Elevation is then first linearly interpolated inside the two chosen DTB segments to yield values exactly at their intersections with the cross-section. Then, elevation is interpolated linearly along the cross-section itself to yield the final elevation of the NWB vertex or densified temporary vertex.

The angle condition mentioned above is a threshold-based evaluation concerning the angle between intersected DTB lines and cross-sections, and is intended to ensure that the chosen DTB segment indeed represents the edge of the selected road, rather than some other feature, or the edge of another road. Hence, the assumption is made that the DTB line segments representing road edges are roughly parallel with the relevant NWB centrelines and lie close to them. Implicitly, this also assumes that NWB centrelines will lie between DTB road edges. In practice, these assumptions are not valid in general, hence a range of failsafe

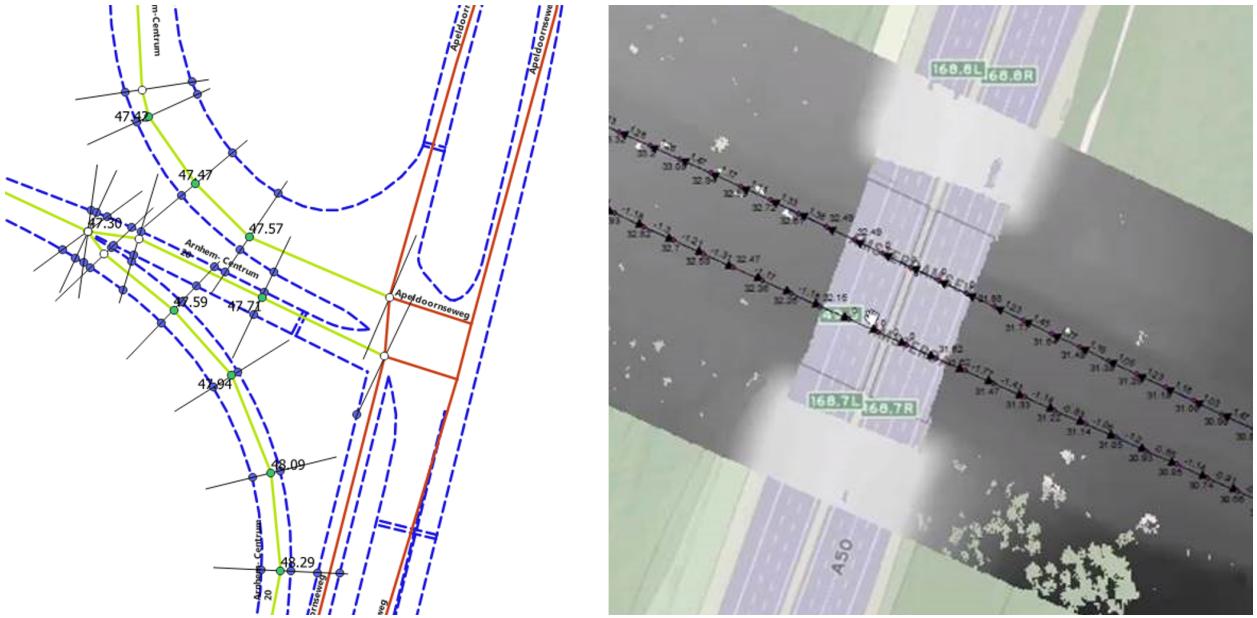


Figure 5.2: Illustrations of RHDHV's commercial implementation. **Left:** The cross-sections that are constructed on NWB vertices are shown as black lines. Green circles denote those vertices where the cross section could be properly intersected with DTB *verflijnen* (blue circles) and thus be given an elevation value. White circles denote where the procedure failed, and AHN raster-based interpolation was necessary. This render illustrates that in sharp bends (where the cross-section might not intersect DTB orthogonally) and close to intersections, this workflow often fails, and that P-roads are not processed in this way (they have DTB edges in this particular location only because they are close to R-roads). **Right:** AHN-based interpolation is used for P-roads, and as a fallback mechanism where DTB-based interpolation fails. The rasters are overlain with NWB to yield elevation values, and as the DSM rasters contain holes, they are patched in by pre-interpolating them before this step. In this illustration, the holes are left in to show that two types of holes generally occur: small-scale ones due to objects such as street furniture, vehicles and vegetation, and large ones that are typically due to occlusion from overlapping buildings or bridges (as in this case). The test build shown here uses AHN2 rasters.

mechanisms needed to be implemented. Wherever the algorithm only finds a suitable DTB intersection on *one* side of NWB, it is only that side from which the elevation value is deduced. If no suitable intersection can be found whatsoever, the AHN3 raster-based interpolation is used instead. At NWB centreline end vertices (where no intersection exists, i.e. dead ends), the previous vertex's elevation is simply repeated. The described methodology is illustrated in Figure 5.2.

It is worth mentioning that the RHDHV implementation deals with all non-standard input data sets (i.e. small-scale engineering models, road management datasets, etc.) by first converting them to rasters with the same specifications as AHN3 DTM tiles and mosaicking them into an output raster based on a priority list, before filling in any remaining gaps and interpolating bilinearly.

From a scientific point of view, there are numerous potential issues suggested by the description of this workflow. For instance, point cloud to raster conversion is, by definition, associated with inherent information loss (less raster cells than pixels), and further reduction in accuracy is introduced by the interpolation mechanism itself. Radial IDW was used to generate AHN3 DTM tiles, which several of the reviewed papers found to be specifically unsuitable for interpolating large-scale areas in which zones of decreased point density or gaps exist – both of which characterise ground-filtered AHN data (e.g. Guo et al. [2010]). In addition, the procedure performs another layer of interpolation to infill gaps, which may further deteriorate accuracy. Furthermore, RHDHV uses bilinear interpolation inside the raster to produce NWB elevations, which is suggested by Shi et al. [2005], to be less accurate than other common methods such as bicubic. In terms of their strong prioritising of DTB for R-roads, I should remark that DTB itself is also a secondary source of information (it is based on a procedural combination of data from various types of sensing, into vector features), and in contrast with AHN3, neither its overall nor its local accuracy are known. While it no doubt contains valuable information, relying on it as the sole source of elevation data does not appear to permit the estimation of output accuracy, which is a pre-requisite of compliance with SWUNG2.

5.3 First approximation of methods to be implemented

The proposed first approximation of the exact methods is the result of a combined understanding of concepts described in related work, my own knowledge and experience relating to the geomatics discipline, as well as inspiration from the commercial implementation of RHDHV. The proposed workflow was built with the research questions in mind. The below summary is only a brief overview, as the detailed specifications will be refined iteratively during development.

1. Pre-processing

- a) Keep only NWB R-roads and P-roads. Identify *non-branching road segments*, henceforth referred to as NBRSs. To identify them, first look for interconnected networks of MultiLineString objects sharing the same street name, then split off branches at intersections, always doing so in the order of decreasing angle (to ensure that a straight continuation of roads is natively preferred). Perform vertex densification for NBRS edges longer than a set distance.
- b) Keep only AHN3 points within a set distance from NWB lines. Keep classes 2 and 26 only.
- c) Keep only DTB “verflijnen”.

2. Point cloud partitioning

- a) For each NBRS from the previous stage, fit planes to the neighbourhood of edges and fetch points that are close to them. When selecting a final plane from candidate planes, perform a similarity check between the parameters of neighbouring planes, so that the resulting succession of planes are not oriented unrealistically with respect to each other. Determine the exact procedure and parametrisation in a way that it is not too complex computationally, but still captures most Lidar points relevant to the given NBRS (i.e. do not use conservative thresholds at this stage).
- b) Merge the sub-clouds of edges into a single sub-cloud for the NBRS and save to disk with an identifier linking it to the NBRS itself.

3. Road edge identification

To be performed on each generated NBRS (and linked sub-cloud of Lidar points).

- a) Construct cross-sections on NBRS vertices (including densified vertices) and snap close-by AHN3 points to them at a pre-set sampling distance along each of them.
 - i. Perform linear regression in their elevation profiles and discard non-conformant points.
 - ii. Disregard points separated by gaps (created by the previous step) from the main group of points representing the fitted line (close to NWB), and points outside a maximum allowed road width. The outermost points in the detected series represent the approximate local position of the road edge on each side.
 - iii. Disregard cross-sections where steps i. or ii. indicate that locally, NWB does not lie on the road surface as suggested by AHN3 (for instance, because the cross-section regression line does not cross the centreline in 2D).
 - iv. Derive mean elevations for each cross-section from the remaining points.
- b) The series of mean elevations (one per cross-section) is itself a 1D elevation profile. Perform outlier detection by sliding a kernel along this profile. Discard cross-sections where this operation indicated that the fitted line is significantly above the road surface. This will help eliminate cross sections corrupted by local groupings of non-road points (such as class-26 motorway signs).
- c) Assemble approximate global road edges from the two outermost Lidar points (on each side of NWB) of each cross-section kept after the previous step.
- d) Use the left and right road edge estimates from the previous step as initial approximations in an active contour optimisation step. The constraints (energy terms): realistic horizontal distance from NWB for Dutch roads, and/or realistic distance from its own initial edge estimate, and a term “detecting” the first noticeable local change in curvature away from NWB.

- e) Select the Lidar points that lie between the optimised road contours in 2D. Thin the selected points so that only the minimum point density remains that is needed in terms of accuracy.
- f) Insert the optimised contours into a CDT as constraints, and then the thinned Lidar points. Before each Lidar point insertion, interpolate in its location in the pre-existing CDT and compare the interpolated elevation to that of the Lidar point to make certain that it does not introduce unwanted curvature into the TIN. Conservative thresholds are appropriate at this stage, as road surfaces are expected to be flat locally.
- g) Interpolate NWB in the CDT using linear, Laplace or natural neighbour interpolation (or some specialised variation thereof that uses a larger query zone and not just a single cell of the tessellation).

4. NBRS merger

- a) Re-assemble NWB from the NBRSs.
- b) Corrections at NWB intersections, smoothing. Yet undetermined if this will be necessary, as it depends on the final implementation. This topic is discussed below in more depth.

5. DTB-based filling of large data gaps (more on this later in this section)

Avoiding the creation of sudden jumps where NBRSs are stitched together may require special attention. In view of the above workflow, we may observe that *in areas surrounding intersections*, the CDT of each NBRS terminating there (or crossing it) will be constructed from roughly the same set of Lidar points (the road points forming the real-life intersection). This is supported by the fact that the CDT construction step is mostly independent of the preceding cross-section-based workflow and the active contour approximation in the sense that it works directly from the Lidar data. However, the proposed segmentation workflow inhibits the creation of identical CDTs at intersections because the edge-based selection procedure may not select the *exact* same Lidar points for each NBRS that terminates at or crosses the intersection. This means that the same intersection vertex may be given different elevation values in different NBRSs, giving rise to ambiguity.

Furthermore, depending on the quality of the raw output elevation profiles, some form of constrained smoothing (or other form of post-processing) may also be needed *generally*, not only at intersections. One solution that is applicable to both purposes in theory, is spline fitting. Using NWB vertices as control points could ensure not only continuity across intersections, but formal C^1 smoothness across them and C^2 everywhere else. However, this raises the question of how we should then treat the rule that the lateral position of NWB centrelines should not change (i.e. in theory the spline fitting would only have one degree of freedom).

Unfortunately, since the above procedure is specialised to reconstructing 3D road geometries within their paved extents, it is not directly applicable to the additional requests I received from NDW regarding the enrichment of lines with elevation data to represent the *vicinity* of roads. The specification of this request included the condition that the exact same methodology needs to be employed in the enrichment of these lines, as the one used for the centrelines themselves. Our CDT road models will not extend beyond the edges of roads, hence it will be impossible to use them to convert lines to 3D that are not on road surfaces. To make this possible, the CDT would need to be *extended* by inserting ground points outside the road edge constraints, beyond where the vicinity-line (effectively buffered centreline) would lie. This would require working outside the segmented point clouds of NBRSs and fetching points directly from AHN3 tiles for a second time. As this direction of research does not fit into my methodology well and given that its motives are not strongly scientific, it is not clear at this point whether it will be tackled.

The above workflow uses concepts already discussed in the last few paragraphs in the Related work section (2). For instance, the point cloud segmentation to decompose the problem into 2.5D sub-problems was inspired by [Oude Elberink and Vosselman \[2009\]](#) and [Boyko and Funkhouser \[2011\]](#). The cross-section based workflow was, among others, inspired by [Yang et al. \[2013\]](#) and the commercial implementation. The use of a CDT to represent the final surface comes from [Oude Elberink and Vosselman \[2006\]](#). The active contour-based workflow was inspired by [Boyko and Funkhouser \[2011\]](#) and [Göpfert et al. \[2011\]](#). However, while in previous research contours were snapped to road curbs, my road-edge energy term will not be specialised to traditional curb geometries. It will be more general; a term that attracts the contour to the first noticeable change in curvature away from NWB.

For the accuracy assessment part of the project, I propose the following secondary workflow:

1. Pre-processing

The accuracy of all input datasets is unaffected.

2. Point cloud partitioning

The point density and spatial distribution of Lidar points decreases. However, these aspects will be considered in later steps, hence it is not necessary to quantify them here.

3. Road edge identification

The main workflow ensures that interpolation takes place in a TIN generated from raw Lidar points. In view of this, the main aspects that need to be examined, in decreasing order of expected importance:

- a) Local controls on accuracy (mainly point density and distribution, curvature). The distribution of the points (e.g. elevation variance) will be examined as an indicator of how successful the algorithm was in selecting road points only. As the stock ground filtering of AHN3 plays a part in this, the local distribution of the points will be considered indicative of that too.
- b) Interpolation accuracy. The two planned approaches are running Monte Carlo simulations on the final interpolator to see how input errors propagate through it and what factors affect it the most, and interpolating in the locations of Lidar points that lie between the road edges, but were not selected to be part of the CDT. Surveying control points is *not* planned to be part of the accuracy assessment procedure.
- c) For each output vertex on NWB centrelines, local CDT vertex elevation variance, position relative to road edges and local road width will be recorded. Together, these will be indicative of how flat the road is between the contours, as well as how successful the algorithm was at pinpointing road edges and how well that agrees with the NWB centreline. Based on this, it will be possible to detect areas where the procedure failed or performed very poorly due to inaccuracies in the position of NWB or the optimised edges, or for other reasons.
- d) For the same reasons as in c. above, road point labelling completeness will be estimated manually while fine-tuning the contour optimisation workflow. This will be based on drawing *approximate* road polygons on AHN3 rasters and overlaying them with the *optimised* polygons (assembled from the optimised contours). I may examine whether BRT road polygons can be used as a reference when estimating completeness over larger areas.

4. NBRS merger

- a) The accuracy description of vertices that are part of multiple NBRSs (i.e. intersection vertices) will need to be aggregated, in the same way as the elevations themselves are aggregated. Instead of aggregating, it may prove to be more effective to simply pick the intersection elevation that has the highest estimated accuracy, and disregard its less accurate counterpart(s).
- b) In case any form of smoothing or other post-processing is implemented, it will need to be possible to control how much it can adjust elevation values (to avoid moving outside the elevation uncertainty range of a predefined threshold).

Due to the completeness problems, topological issues, and unverified accuracy of DTB as described in the Datasets and tools section (4), I did not include it in the primary workflow. Mostly because of the latter, from a purely theoretical point of view, its elevation values cannot be used in a way that influences derived elevations, because that would prevent the estimation of output accuracy.

However, there might be other uses for it that do not have this side-effect. Firstly, a DTB *verflijn* is generally found close to the edges of most R-roads and as a result, they may be useful as fallback road edge estimates wherever the 1D line-fitting-based method fails, in which case they would need to be intersected with the cross sections in 2D, much like in the commercial implementation. Secondly, in most places DTB appears to consistently represent the lines that are painted on R-roads a fixed distance from the actual edges of the asphalt. As such, where they do indeed represent these lines, they could be useful as secondary attractors in

the active contour optimisation step, perhaps as a safeguard mechanism to ensure that blunders in the cross-section based initial road edge approximations do not affect the final road contours too badly. Furthermore, based on my preliminary analysis, the lateral location of DTB lines tends to agree better with AHN3 than that of NWB. Hence, NWB's position relative to the closest DTB lines on each side could be a good indicator of local lateral inaccuracy in NWB. In addition, on bridges (where class 26 also contains the supporting structures of bridges), DTB lines may provide important first approximations of the road surface plane.

Lastly, but perhaps most importantly, DTB can be used to fill large Lidar data gaps such as those appearing underneath big structures covering the surfaces of R-roads, and inside tunnels. Assuming the accuracy assessment workflow is implemented in a similar way as in the description above, then these locations will be characterised by extreme drops in point density, anomalous point distribution, unusually large CDT triangles, and as a result, low interpolation accuracy. In other words, we can use the derived accuracy to detect where the algorithm encountered data gaps, and estimate how big the gaps are. For locations where only a few vertices are missing (e.g. a length of road covering only 10-20 metres), linear interpolation inside the elevation series is probably reasonable, although it will need to be indicated in the output where this has taken place. Alternatively, the original interpolated values can be left in, assuming they are not outliers. However, where many such vertices are found in a succession, the program must assume the presence of a large AHN3 data gap. In such areas, DTB could be used as a fallback source of elevations, as it was augmented with land-based survey data wherever its primary photogrammetry-based workflow yielded insufficient data (thereby containing useful data inside tunnels and under occluding objects). However, its use would need to also be marked semantically in the output because of the unknown accuracy of DTB.

To make such uses of DTB possible, it would first need to be pre-processed so that only the correct, edge-representative DTB lines remain (not, for instance the stop lines shown in the top left image in Figure 4.5). Furthermore, smoothing may need to be performed close to where DTB-based elevations are "patched into" the AHN3-based interpolation results – as the commercial results have already indicated, there can be significant differences between elevations suggested by DTB and AHN3 for the same section of a road.

Given the trial-and-error nature of implementing these additional DTB-based workflows in the overall procedure, some of them may end up in the final implementation, while others may be omitted. This is the reason why they are not included in the formal workflow description and are instead only vaguely described in this paragraph.

Testing the accuracy of the commercial implementation will take place via two different approaches. The ideal approach, which would be to enable the computation of the formal accuracy inside the commercial application by injecting additional code, are made difficult, likely impossible, by two factors. Firstly, their code is written in *ArcPy*, hence the first step would be to port their entire codebase into the open-source framework that my implementation will be built in, or at least the main algorithms from it. Furthermore, this approach could never yield accuracy values for R-road elevations, because RHDHV rely on DTB for these roads, which does not have a formal accuracy description in its documentation. As a result, attempting this approach is not well justified.

Hence, my first method will involve merely examining general properties of the output, such as smoothness, density of outliers and missing values. It will also involve a visual assessment of their results, including comparisons with the AHN3 point cloud and my own results, particularly in difficult environments. The second approach will involve deriving errors and RMSE values *relative* to my own results. Making this comparison will indicate where the commercial results diverge from the ranges of plausible values, as specified by the uncertainty ranges in my output. Where their output falls outside these ranges of uncertainty – which could be examined visually by plotting the differences on NWB centrelines 2D – I will closely examine the two results in the context of differences in the methodologies and local features in the data sources, and attempt to explain the disagreement scientifically.

6 Project Schedule

The following page contains a diagram illustrating the schedule of my graduation year. Please note that the exact P3, P4 and P5 dates are not yet known, hence the week in which they are shown in the diagram are tentative. As a result, the overall length of time available in the second semester is also not yet fixed; the longest possible time is shown.

Graduation year schedule

References

- Aguilar, F., Agüera, F., M.A., A., and Carvajal, F. (2005). Effects of terrain morphology, sampling density, and interpolation methods on grid dem accuracy. *Photogrammetric Engineering and Remote Sensing*, 71:805–816.
- Aguilar, F., Mills, J., Delgado, J., M.A., A., Negreros, J., and Pérez, J. (2010). Modelling vertical error in lidar-derived digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65:103–110.
- Bater, C. and Coops, N. (2009). Evaluating error associated with lidar-derived dem interpolation. *Computers Geosciences*, 35:289–300.
- Bell, M. and Lida, Y. (1997). *Transportation network analysis*.
- Bennett, J. (1997). The application of geographic spatial analysis in modeling traffic noise propagation away from a freeway. *Journal of The Acoustical Society of America*, 102.
- Boyko, A. and Funkhouser, T. (2011). Extracting roads from dense point clouds in large scale urban environment. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66.
- Cai, H. and Rasdorf, W. (2008). Modeling road centerlines and predicting lengths in 3-d using lidar point cloud and planimetric road centerline data. *Computer-Aided Civil and Infrastructure Engineering*, 23:157–173.
- Caltagirone, L., Scheidegger, S., Svensson, L., and Wahde, M. (2017). Fast lidar-based road detection using fully convolutional neural networks. pages 1019–1024.
- Chen, C.-F. and Hsu, C.-J. (2020). Using a gis road network to improve gps positions.
- Chow, E. and Hodgson, M. (2009). Effects of lidar post-spacing and dem resolution to mean slope estimation. *International Journal of Geographical Information Science*, 23:1277–1295.
- Clode, S., Kootsookos, P., and Rottensteiner, F. (2004). The automatic extraction of roads from lidar data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35.
- Clode, S., Rottensteiner, F., Kootsookos, P., and Zelniker, E. (2007). Detection and vectorization of roads from lidar data. *Photogrammetric Engineering and Remote Sensing*, 73:517–535.
- Durán Fernández, R. and Santos, G. (2014). A gis model of the national road network in mexico. *Research in Transportation Economics*, 46.
- Ekpenyong, F., Brimicombe, A., Palmer-Brown, D., Li, Y., and Lee, S. (2007). Automated updating of road network databases: road segment grouping using snap-drift neural network. *Proceedings of Advances in Computing and Technology*, pages 160–167.
- Fouque, C. and Bonnifait, P. (2008). Tightly coupled gis data in gnss fix computation with integrity test. *Journal of Intelligent Information and Database Systems*, 2:167–186.
- Gross, H. and Thoennesen, U. (2006). Extraction of lines from laser point clouds. *Fraunhofer FOM*, 36.
- Guarnaccia, C. and Quartieri, J. (2012). Analysis of road traffic noise propagation. *International Journal of Mathematical Models and Methods in Applied Sciences*, 6:926–933.
- Guo, Q., Li, W., Yu, H., and Alvarez, O. (2010). Effects of topographic variability and lidar sampling density on several dem interpolation methods. *Photogrammetric Engineering and Remote Sensing*, 76.
- Göpfert, J., Rottensteiner, F., and Heipke, C. (2011). Using snakes for the registration of topographic road database objects to als features. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66:858–871.
- Hatger, C. and Brenner, C. (2003). Extraction of road geometry parameters from laser scanning and existing databases. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34:225–230.
- Hodgson, M. and Bresnahan, P. (2004). Accuracy of airborne lidar-derived elevation: Empirical assessment and error budget. *Photogrammetric Engineering and Remote Sensing*, 70:331–339.
- Hu, X., Tao, C., and Hu, Y. (2004). Automatic road extraction from dense urban area by integrated processing of high resolution imagery and lidar data. volume 35.
- Hu, Y. (2003). *Automated extraction of digital terrain models, roads and buildings using airborne lidar data*. PhD thesis, University of Calgary.
- Ishiyama, T., Tateishi, K., and Arai, T. (1991). An analysis of traffic noise propagation around main roads in tokyo. *Noise Control Engineering Journal*, 36.
- Kraus, K., Karel, W., Briese, C., and Mandlburger, G. (2006). Local accuracy measures for digital terrain models. *The Photogrammetric Record*, 21:342 – 354.

- Lin, Y., Wang, C., Cheng, J., Chen, B., Jia, F., Chen, Z., and Li, J. (2015). Line segment extraction for large scale unorganized point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 102.
- NDW (Accessed 09/12/2020). NWB is Geometrisch Gecorrigerd. <https://nationaalwegenbestand.nl/nieuws/nwb-geometrisch-gecorrigeerd>.
- Oude Elberink, S. (2010). *Acquisition of 3D topography: automated 3D road and building reconstruction using airborne laser scanner data and topographic maps*. PhD thesis, University of Twente.
- Oude Elberink, S. and Vosselman, G. (2006). Adding the third dimension to a topographic database using airborne laser scanner data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36.
- Oude Elberink, S. and Vosselman, G. (2009). 3d information extraction from laser point clouds covering complex road junctions. *Photogrammetric Record*, 24:23–36.
- Oude Elberink, S. and Vosselman, G. (2012). Quality analysis of 3d road reconstruction. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36.
- PDOK (Accessed 09/12/2020). AHN3 Downloads. <https://downloads.pdok.nl/ahn3-downloadpage/>.
- Peng, J., Parnell, J., and Kessissoglou, N. (2020). Spatially differentiated profiles for road traffic noise pollution across a state road network. *Applied Acoustics*, 172.
- Peng, M.-H. and Shih, P. T.-Y. (2006). Error assessment in two lidar-derived tin datasets. *Photogrammetric Engineering Remote Sensing*, 72:933–947.
- Raber, G., Jensen, J., Hodgson, M., Tullis, J., Davis, B., and Berglund, J. (2007). Impact of lidar nominal post-spacing on dem accuracy and flood zone delineation. *Photogrammetric Engineering Remote Sensing*, 73:793–804.
- Rijkswaterstaat (2020b). Handboek DTB.
- Rijkswaterstaat (2020c). Handleiding Nationaal Wegenbestand (NWB-Wegen).
- Rijkswaterstaat (2020d). Productspecificaties Digitaal Topografisch Bestand.
- Rijkswaterstaat (Accessed 09/12/2020a). AHN Kwaliteitbeschrijving. <https://www.ahn.nl/kwaliteitsbeschrijving/>.
- Shi, W., Li, Q., and Zhu, C. (2005). Estimating the propagation error of dem from higher order interpolation algorithms. *International Journal of Remote Sensing*, 26:3069–3084.
- Smith, S., Holland, D., and Longley, P. (2005). Quantifying interpolation errors in urban airborne laser scanning models. *Geographical Analysis*, 37:200 – 224.
- Su, J. and Bork, E. (2006). Influence of vegetation, slope, and lidar sampling angle on dem accuracy. *Photogrammetric Engineering Remote Sensing*, 72:1265–1274.
- Taylor, G., Blewitt, G., Steup, D., Corbett, S., and Car, A. (2001). Road reduction filtering for gps-gis navigation. *Transactions in GIS*, 5:193–207.
- Thomson, R. and Richardson, D. (1995). A graph theory approach to road network generalisation. *Int. Cartographic Conference*, pages 1871–1880.
- Vosselman, G. and Liang, Z. (2009). Detection of curbstones in airborne laser scanning data. *Remote Sensing and Spatial Information Sciences*, 38.
- Wang, J., Lawson, G., and Shen, Y. (2014). Automatic high-fidelity 3d road network modeling based on 2d gis data. *Advances in Engineering Software*, 76:86–98.
- Yang, B., Fang, L., and Li, J. (2013). Semi-automated extraction and delineation of 3d roads of street scene from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 79:80–93.
- Yue, Y., Yeh, A., and Li, Q. (2008). Road network model for vehicle navigation using traffic direction approach. pages 613–629.
- Zhang, H. (2011). *Uncovering road network structure through complex network analysis*. PhD thesis, Hong Kong Polytechnic University.
- Zhang, W. (2010). Lidar-based road and road-edge detection. pages 845 – 848.
- Zhu, L. and Hyppä, J. (2014). Fully-automated power line extraction from airborne laser scanning point clouds in forest areas. *Remote Sensing*, 6(11):11267–11282.
- Zhu, Q. and Li, Y. (2007). Review of road network models. pages 471–476.
- Zhu, Q. and Li, Y. (2008). Hierarchical lane-oriented 3d road-network model. *International Journal of Geographical Information Science*, 22:479–505.
- Zhu, Q. and Mordohai, P. (2009). A minimum cover approach for extracting the road network from airborne lidar data.