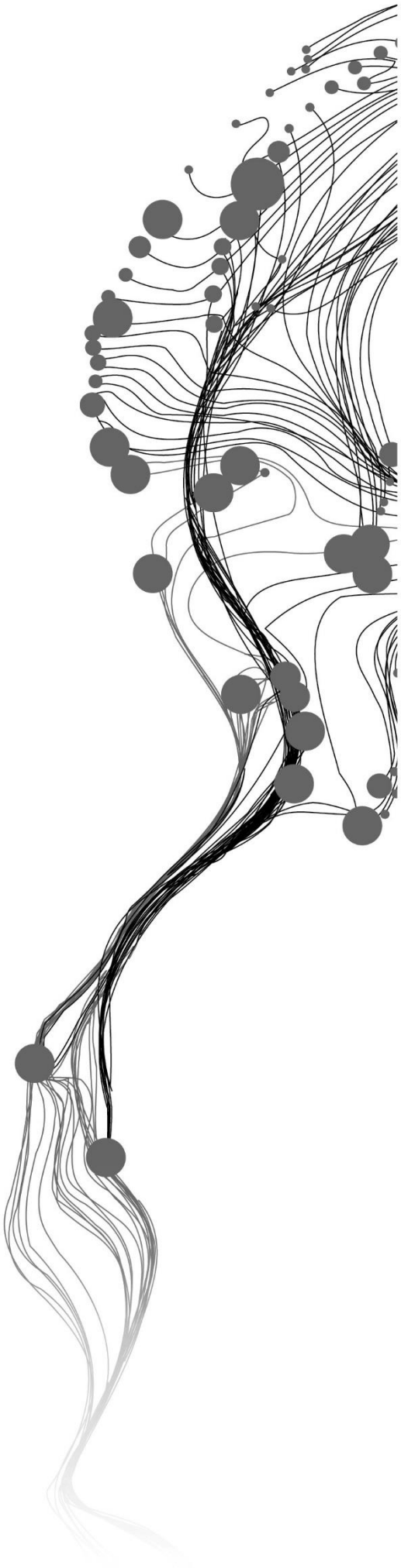
ASSESSING THE FLANGEWAY OF RAILWAY SWITCHES USING POINT CLOUDS CAPTURED FROM A MOVING TRAIN

ANNA ENNIN
June 2021

SUPERVISORS:
Dr. V. V. Lehtola
Dr. Ir. S.J. Oude Elberink



ASSESSING THE FLANGEWAY OF RAILWAY SWITCHES USING POINT CLOUDS CAPTURED FROM A MOVING TRAIN

ANNA ENNIN
Enschede, The Netherlands, June 2021

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.

Specialisation: Geoinformatics

SUPERVISORS:
Dr V. V. Lehtola
Dr Ir. S.J. Oude Elberink

THESIS ASSESSMENT BOARD:
Prof. Dr Ir. M.G. Vosselman (Chair)
Dr Juho-Pekka Virtanen, Aalto University, Dept of Built Environment
Dr V. V. Lehtola
Dr Ir. S.J. Oude Elberink

ABSTRACT

Railway switches are the nodes of the track network and therefore must operate with high reliability. Switches are worn down by normal operations because they contain moving parts and experience tremendous forces from the trains they are directing. It is highly important to detect this wear before a train becomes derailed. However, visual assessment and manually measurement with tapes serve as the fundamental approach to obtain the parameters used to assess these switches. Mobile mapping systems have become a popular means for railway infrastructure companies to acquire data that are used for various track monitoring purposes. This research, therefore, presents the feasibility of obtaining one of the switch inspection parameters from a 3D point cloud captured with a mobile mapping system. One important assessment parameter of railway switches is the flangeway, which is the minimum distance between the stock and switch rail when the switch is opened. Finding this minimum distance rigorously requires continuous modelling for the rails. Hence, this study presents an algorithm that turns the discrete point clouds into continuous models to find the flangeway. First of all, the rails are extracted from the point clouds using segmentation. Direct measurement on the rail point clouds may result in incorrect determination of the flangeway. Therefore, we resort to determining the positions (running edge points) on the rails where the flangeway is measured from the point clouds using prior knowledge of the designed rail profile. Due to the different curvature of the rails, a B-spline curve is fitted to the running edge points to generate continuous models for the stock rail and switch rail. Finally, the models are used to compute rail-to-rail distances in a continuous manner, which allows for the extraction of the minimum distance (the flangeway). The prosed method has proven its ability to determine switch flangeway from point clouds and can be used for commercial purposes. Also, most of the methods are implemented in python and can easily be adapted by railway infrastructure management companies to determine the flangeway automatically.

Keywords: Railway infrastructure, 3D point clouds, Railway switches, Flangeway, B-spline curve, Mobile mapping systems.

ACKNOWLEDGEMENTS

My God, who started with me throughout the journey of my MSc education, has surely brought it to an expected end. I am very grateful to God for His uncommon favour and grace to write my thesis to perfection. May His name be praised. Amen!

My sincere appreciation goes to my first supervisors, Dr Ville V. Lehtola and Dr Ir. S.J. Oude Elberink, for their tireless effort, selfless supervisions and helpful criticisms in shaping my research. I would not have made it without your professional, academic help and knowledge shared with me. I am very grateful.

I appreciate the Fugro RailData team, Chris Sullivan, Yi Ding, Haoyu Wang and Pete Apostle for providing me with the data and technical assistance needed to complete my thesis.

I am highly indebted to the Dutch Government for their financial support through the Orange Knowledge Programme (OKP) scholarship awarded to me. Through your support, I have been able to advance my studies, making me more resourceful.

I am grateful to my parents, Janet Forson, Anthony Amissah and John K. Annin, for their prayers and support that kept me going.

Words cannot express how I appreciate my two elder brothers, Patrick Justice Kojo Ennin and Benjamin Arthur, for their constant encouragement, motivation and financial support. I could make it because you guys always had my back and were edging me on. God richly bless you in all you do.

My special gratitude goes to the Ghanaian family Eunice, Leticia, Efia, Adwoa, Mavis, Rexford, Prince, Derrick and Yidanna for their friendship and support throughout my stay in the Netherlands, most especially during my research phase. Every moment spent with you guys has become memories I cannot forget.

Special thanks to Ransford Nii Aryitey Welbeck (King Loco) for your selfless support. I am grateful for your friendship and believing I could do it when I had even given up on myself. You are a rare GEM. I am very grateful.

To Perpetual Akwensi, Richard Aborah, Darshana and Vasudha, thank you for the academic advice and contributions you made toward my thesis. To everyone that contributed to making this journey (Master's) a smooth one, I say "NYAME YIRA WO".

It will surely end in praise. Indeed this is for his will and my good.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. Background and motivation

Travelling by rail has become a popular and preferred means of transportation, which indicates an increase in demand to provide safe and efficient railway operations. Consequently, railway companies perform frequent inspections on various railway infrastructure to improve the various railway practices to meet the growing demand (Rusu, 2015). During these inspections, geographic data on the various infrastructure are gathered for efficient railway asset management and proper maintenance (Lou et al., 2018). The railway infrastructures which receives more attention, and therefore monitoring and maintenance is carried out frequently on them are the railway tracks (Díaz Benito, 2012). This is because irregularities in the tracks affect railway operations, traffic flow, and the general safety and comfort of passengers (Oude Elberink, Khoshelham, Arastounia, & Díaz Benito, 2013). Therefore, the railway tracks are to be inspected regularly to detect any irregularities to schedule frequent and efficient maintenance. The use of traditional railway inspection techniques whereby operators perform field inspections using contact instruments are expensive, time-consuming, and sometimes inefficient (Lou et al., 2018). These techniques require surveyors to use contact instruments for onsite inspection of the railway tracks (Lou et al., 2018). Using these traditional inspections interrupts normal railway operations and further put surveyors at risk. In addition, different parts of the rail geometry require different instruments, and their measurement is solely dependent on the operator's criteria (Díaz Benito, 2012). Semi-automated methods where optical sensors, such as cameras and videos, are used also depend on good lighting and environmental conditions (Lou et al., 2018).

There is a need for an efficient track inspection method for frequent monitoring and maintenance. Mobile Mapping Systems (MMS) has become a viable inspection method, and it is widely used to acquire data on railway tracks for inspection and monitoring (Soni, Robson, & Gleeson, 2014; Karunathilake, Honma, & Niina, 2020). The MMS is mounted directly on the train to inspect the railway environment periodically (Oude Elberink & Khoshelham, 2015; Pastucha, 2016). MMS consist mainly of navigational and data acquisition sensors. The navigational sensors are Global Navigation Satellite Systems (GNSS) and Inertial Measuring Units (IMU) which record the position and orientation of the train as it moves (Mikrut et al., 2016). The data acquisition sensors consist of laser scanners and digital cameras. While the laser scanners are used to acquire the spatial location of the object (in this case, the rail), the digital cameras provide images and videos that contain colour information of the scanned object (Mikrut et al., 2016). This method is now popularly used for railway inspection because it is fast, cost-effective, and independent of lightning conditions (Karunathilake et al., 2020; Arastounia, 2016). It also provides a large amount of accurate 3D data (point clouds), measuring up to a millimetre level accuracy (Karunathilake et al., 2020). The data obtained from the MMS can be used to assess the geometry of the tracks for efficient maintenance and management. Figure 1-1 shows an MMS mounted on the train for acquiring data on objects in the railway environment.

Figure 1-1: A mobile mapping system mounted on a train to acquire data on railway tracks (Source: (Rusu, 2015)).

Railway switches are unique parts of the rail network that guide trains to change from one track to another (Kaewunruen, 2014). They are an essential component of tracks and are required to operate with high reliability; hence they must be well maintained to enhance the safety of the railway network (Hamadache et al., 2019; Rusu, 2015). Switches have a complex design, unlike the plain lines of the tracks (Kaewunruen & Lian, 2019; Rusu, 2015). According to Rusu (2015), switches consist of three (3) main components: switch, crossing, and point machine. The switch component helps the train when changing tracks, and it consists of switch rails and stock rails. The crossing contains wing rails that are responsible for supporting the train wheel at the rail's intersection region. The point machine moves and locks the switch rail as the train travels on it. Figure 1-2 shows the schematic diagram of a simple switch where the various components are labelled. The switch component is mostly exposed to high forces, which result in damages. These high forces are a result of train movement and the frequent movement of the switch rail whenever trains pass on them (Kaewunruen & Lian, 2019; Rusu, 2015). Rusu (2015) stated that due to the frequent damages of switches, the International Union of Railways (UIC), in their report, mentioned that the maintenance of switches cost about 25% to 30% of the total maintenance budget per year. Also, damages in any of the components will result in failure of the whole system, which can cause significant problems such as discomfort and even train derailment (Kaewunruen & Lian, 2019; Bemment, Ebinger, Goodall, Ward, & Dixon, 2017). Therefore, there is the need for frequent inspection of switches for effective maintenance and management to prevent accidents and to reduce the budget used for their maintenance.

For the maintenance of switches, inspection parameters are used to monitor the condition of the switches, which helps determine the damage for necessary actions to be taken (Rusu, 2015). Most of the parameters used to assess the switch are stated in Rusu (2015). There are several techniques used to obtain these inspection parameters of the switch (Turabimana & Nkundineza, 2020; Rusu, 2015). However, visual inspection and manual measuring instruments are the most common techniques used (Turabimana & Nkundineza, 2020; Bemment et al., 2017; Rusu, 2015). Therefore, the motivation of this study to develop a method that can determine one of the switch inspection parameters from point clouds.

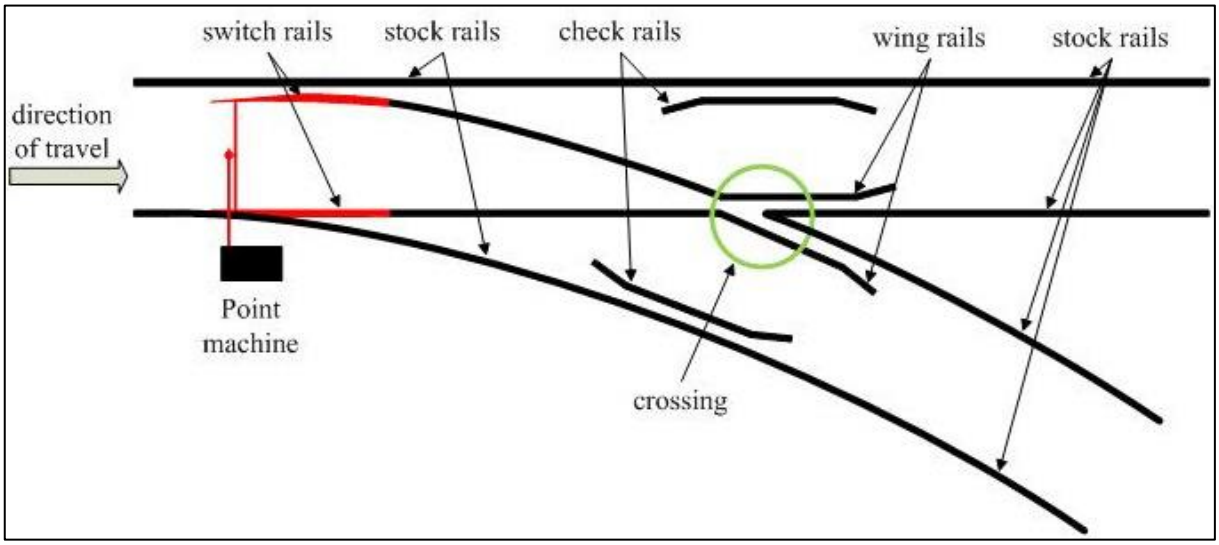


Figure 1-2: Diagram of a common switch (Source: (Rusu, 2015))

Figure 1-3 An operator measuring the distance between the stock rail and its switch rail manually. This is a manual measurement in contrast to what is done in this work from a point cloud (Source: Fugro Raildata).

1.2. Problem statement

An important switch inspection parameter is the switch flangeway, according to industry experts. Switch flangeway is defined as

the minimum distance between the inner running edge of the stock rail and its corresponding switch rail when the switch is opened (Standards British, 2003).

A switch is said to be opened when the switch rail is not attached to the corresponding stock rail (Standards British, 2003). The flangeway allows the wheels of the train to move unhindered while the train changes tracks. When the flangeway becomes narrower, it will cause the wheel to touch the switch rail while the train moves, and this may result in a break in the switch rail (International Union of Railways [UIC], 2015). This effect in the flangeway occurs when the point machine, which is the locking and moving device of the switch, is not adjusted correctly (UIC, 2015). For this study, a method is proposed to determine the switch flangeway from point clouds.

1.3. Research objective

The main objective of this research is to develop an algorithm that can create continuous models from point clouds to determine the switch flangeway. The study, therefore, attempts to find the minimum distance between the stock rail and its switch rail by first separating the rails points. The rail points are then represented with continuous models so that the distances between the rails can be computed in a continuous manner to determine the flangeway. The scanner system used to acquire the point clouds captured the switch such that the inner part of the switch rail is not shown. Therefore, adjustments are made on the measurement objective, whereby instead of directly measuring the flangeway, the distance is measured at the inner running edge of the stock rail to the outer running edge of the switch rail (Figure 1-4). And then, if this method is to be used for commercial purpose, then one has to subtract a constant distance value that is already known from the switch rail profile design schematics to obtain the actual flangeway.

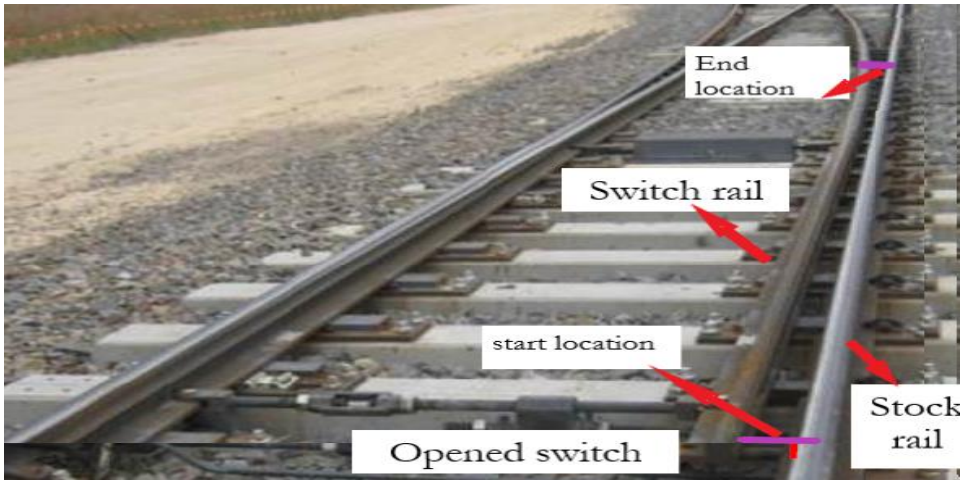


Figure 1-4: Image depicting an opened and region where distances will be computed

1.4. Sub-objectives

- To extract the stock and switch rail as separate rails from the point clouds
- To determine the running edge positions on the rails.
- To determine the flangeway between the stock rail and switch rail

1.5. Research questions

- What method can be used to extract the stock rail and switch rail as separate rails?
- How will the running edge positions of the rails be determined?
- Can distances be determined in a continuous manner between the two rails?
- Does the method work well to determine the flangeway?

1.6. Thesis structure

This thesis is structured as follows. Chapter one presents the introduction, problem statement and objective of the research. Chapter two provides literature on the theoretical principles and some previous studies related to this research. Chapter three describes the data and software used for testing the proposed methodology. In Chapter four, the research workflow and proposed methodology are explained. The results obtained after testing the methodology are presented in Chapter five. It describes the results according to each research sub-objectives. Chapter six discusses the main findings. This is followed by Chapter seven, where the thesis is concluded, and recommendations for future work are outlined.

2. LITERATURE REVIEW

This chapter reviews selected literature on some of the processes used to achieve the objective of this research. The section begins with literature on various point clouds segmentation methods.This is because the study seeks to use a segmentation approach to extract the rails. This is followed by previous studies on the extraction of railway tracks from point clouds. Lastly, as the rail switches experience deformation, which this study seeks to determine, we review selected literature on various deformation monitoring using point clouds.

2.1. Segmentation of point clouds

Point cloud segmentation refers to the grouping of points with comparable radiometric, geometric, colormetric and any other essential characteristics to boost the processing of point cloud data (Che & Olsen, 2018). The results of point cloud segmentation are groups of homogenous points that can be employed in performing operations such as extracting, modelling, classifying and analysing features. Several studies have examined point cloud segmentation. In these studies, four primary geometric based approaches have been documented. These include edge-based, model fitting-based, clustering feature-based and region growing-based approaches.

Edge-based approaches identify borders of areas in point cloud data to group the data into regions. As indicated by Rabbani, van den Wildenberg, and Vosselman (2006), two core tasks ought to be completed when executing the edge-based segmentation approaches. The tasks comprise of detecting the edges and then categorising the points to construct segments. With regards to edge-based approaches, points with fast altering geometric feature are linked to create occlusive segmentation areas (Rabbani et al., 2006). However, the sensitivity to noise and the exhibition of poor performance on intricate scenes are drawbacks of such approaches (Rabbani et al., 2006).

Model fitting-based segmentation approaches are centred on the grounds that natural objects typically can be ordered into unsophisticated geometric elements such as planes, and cylinders, just to mention a few. Thus, point cloud data are fitted to geometric elements, and points with consistent mathematical description are categorised as a single segment. Hough transform and Random Sample Consensus (RANSAC) are the two most applied model fitting-based methods. Although model-based methods fail for unsupervised applications and complex shapes, these methods are fast and robust to outliers (Hesami, BabHadiashar, & HosseinNezhad, 2010). Geometric and radiometric traits are generally utilised as features when segmenting point cloud data with clustering techniques (Wang, Tan, & Mei, 2019). Features are extracted and later clustered when applying clustering feature-based methods (Biosca & Lerma, 2008). Regular and irregularly shaped targets can be processed using clustering feature-based methods, but the quality of the results depends on the selection and computation of the point features (Lu et al., 2016).

The region growing-based methods involve clustering neighbouring points that share analogous attributes into a single element. Owing to a pre-established similarity criterion, a region that can also be referred to as a component or element is grown around a seed point until an end condition is encountered. The region growing-based methods outperform the edge-based methods on continuous surfaces. However, the outcome of applying region growing-based methods is dependent on the pre-established similarity criterion and the seed points used. An enhancement of the region growing-based method known as the connected components can be applied to regular and irregular targets and necessitates no seed point to function(Chen et al., 2019; Trevor, Gedikli, Rusu, & Christensen, 2013). As a result, connected components have been widely used for point clouds segmentation (Belkhouche, Duraisamy, & Buckles,

2015). As the rails are continuous surfaces, it is expected that the connected component -method would be best suited for extracting the rail elements from the point clouds.

2.2. Extraction of rails from point clouds

In the railway environment, past studies have extracted railway objects from point clouds. One of the objects which previous studies have focused on is the railway track (Díaz Benito, 2012). Most of these studies have developed methods to detect and extract railway tracks from point clouds.

For instance, Oude Elberink et al. (2013) used a knowledge-based approach together with the RANSAC algorithm to detect rails. The study employed some characteristics of rail tracks and contact wires as the basis of detection. These characteristics include the relative heights of rail tracks and contact wires, relative positions of rail tracks and contact wires to other objects, linearity of rail tracks and contact wires and relative position of one rail track to another. The results obtained an accuracy of 2 cm. The detected rail points were classified and used in generating 3D models for the rails. Yang and Fang (2014) extracted railway trucks by using a data-driven technique that utilises the geometry and reflective properties of the tracks. The rails are detected as linear objects in the point clouds based on the shape and dimensionality features. The results achieved an accuracy of 95.43%. Another interesting study on railway extraction was conducted by Jwa and Sonh (2015). They propose a method that initially approximates the orientation of the track trajectory. Based on the determined orientation, strips perpendicular to that trajectory are created from the point clouds. For each strip, a Bayesian process was used to detect track regions and points, from which a region growth approach is used to segment rail head points. The rail points were accurately detected with correctness and completeness score of 99.9% and 81.7%, respectively. The points were used in generating track models, which also obtained a Root Mean Squared Error (RMSE) of 1.8 cm. Relying on the shape, geometry, as well as intensity features, Lou et al. (2018) proposed a method to automatically detect railway tracks. The study concluded their method could extract tracks in 3D and also in real-time with an accuracy of about 99.7%. Niina et al. (2018) extracted the railway trucks from point clouds to inspect the clearance along West Japan Railway Company's railway tracks. They employed the iterative critical point (ICP) algorithm to determine rail positions by matching the ideal rail head shape with their respective point clouds. The rail positions were extracted with an accuracy of less than 3 mm. The aforementioned studies obtained very good accuracies in detecting and extracting railway trucks in straight sections; however, in switch locations, these studies were not able to detect points belonging to the rails. This is due to the complex nature of the switch in the point clouds.

Zou et al. (2019) proposed an algorithm that could distinguish between track branches to detect rails in bends and switches. Their algorithm makes use of the track's geometry and intensity characteristics of the point clouds without any prior information on the trajectory of the track. It employs a K-means clustering fused Region-Grow Fitting technique to select track points based on the intensity of the point clouds. Karunathilake et al. (2020) made an improvement to the algorithm proposed by Niina et al. (2018) to detect rails in all sections of the railway. These sections include the linear section, turnout(switches) and crossings. Their detection algorithm required less background information on the railway track structure. The proposed algorithm obtained 100% accuracy for detecting the railway tracks. The results were used to estimate the gauge corners of the detected tracks accurately. Although these studies achieved success in detecting the rails in the switch section of railway tracks, no further analysis was done to determine any of the inspection parameters which are used to assess the condition of these switches.

2.3. Deformation monitoring and change detection using point clouds

Monitoring deformations or changes in the state of objects is an important area of research for many application areas (Mukupa, Roberts, Hancock, & Al-Manasir, 2016). According to Mukupa et al. (2016), deformation monitoring and change detection are related, although they are different terminologies. They defined deformation monitoring as making systematic measurements to track changes in the shape, extent, and position of an object resulting from stress. On the other hand, change detection finds differences in the state of the object by studying it at different epochs (Mukupa et al., 2016). Analysis for deformation monitoring seeks quantitative changes, whiles change detection seeks a binary answer to know whether the objects state has changed or not (Lindenbergh & Pietrzyk, 2015). Although there are a number of studies relating to the aforementioned techniques, those using point clouds have gained more attention (Mukupa et al., 2016; Walton, Delaloye, & Diederichs, 2014).

When using point clouds for deformation monitoring, extracting only the object points is not enough but will require modelling of the object (Bureick, Alkhatib, & Neumann, 2016; Bureick, Neuner, Harmening, & Neumann, 2016). This is due to the individual nature of the points, which may lead to inaccurate results when direct measurement is made on them. Neuner, Holst and Kuhlmann (2016) provide an overview of different approximation and modelling techniques for various objects when using point clouds to derive information on deformation. They stated that, for continuous surfaces, geometric models are mostly employed. Geometric models mostly used for approximating point clouds are the free-form models (Bureick, Neuner, et al., 2016). These models are generated with Bezier, B-spline and Non-uniform rational B-Splines (NURBS) curves. Using these free form models helps to approximate all types of objects since the point clouds of these objects may have data gaps, cusp as well as sharp and irregular edges (Bureick, Neuner, et al., 2016).

Amongst the free-form curves and surfaces, B-splines are often used to approximate objects with complex geometry (Bureick, Alkhatib, et al., 2016; Zhao, Kargoll, Omidalizarandi, Xu, & Alkhatib, 2018). Bureick, Neuner, et al. (2016) conducted a study to compare different point cloud approximation methods for deformation analysis. The models compared were polynomials, Bezier, B-spline, and NURBS. It was concluded that B-splines are suitable for fitting all types of points clouds and can deal with data with gaps. To determine the deformation in an arch structure using point clouds capture by TLS, Zhao et al. (2018) modelled the surface using both B-spline and polynomial approximation methods. They concluded that with appropriate model parameters, the B-spline has a significant advantage over the polynomial model. Other studies have used B-spline to determine deformation in tunnels and railway tracks and reported its robustness to outliers and data gaps (Xu, Yang, & Kargoll, 2019; Bureick, Alkhatib, et al., 2016). For monitoring deformation in railway switches, Rusu (2015) confirmed that visual inspection was mostly used and therefore develop an automatic method using point clouds. The study created a convex hull of the point cloud of the rails, which was used to determine various deformations in the switch. This study seeks to create a continuous model for the stock rail and its switch rail using a B-spline spline curve due to its advantages stated above.

3. DATA AND SOFTWARE USED

3.1. Data

This research data consists of 3D point clouds of railway switches provide by Fugro RailData and acquired with a Rail Infrastructure Alignment Acquisition (RILA) system. The RILA system used by Fugro is a mobile mapping system that is attached to the back of the train to acquire data on railway infrastructure as the train travels. The system consists of a GNSS/IMU, 360 lidar scanner, three (3) video cameras, and two (2) rail scanners, as shown in Figure 3-1 (Fugro, 2021). The data for this study is an output from the two rail scanners; the scanners are positioned on the left and right to acquire data for the individual rails. Data from the rail scanners are basically used for track geometry inspection (Fugro, 2021). The rail scanner consists of two sensors, a laser and a camera. The laser scans the rail by drawing a laser line that produces a series of points across the rail profile while the camera records a high-frequency image of the scanned area. Therefore, the output from the rail scanner is a 2D dense sequence of points of the rail profiles. The 2D points are then transformed to 3D world coordinates using the GNSS/IMU unit positions. The final data output are individual pieces(strip) of 3D points for each rail. A data strip contains points of objects on the track bed, including the stock and switch rails, sleepers, and other objects close to the rail, such as rail supporters.

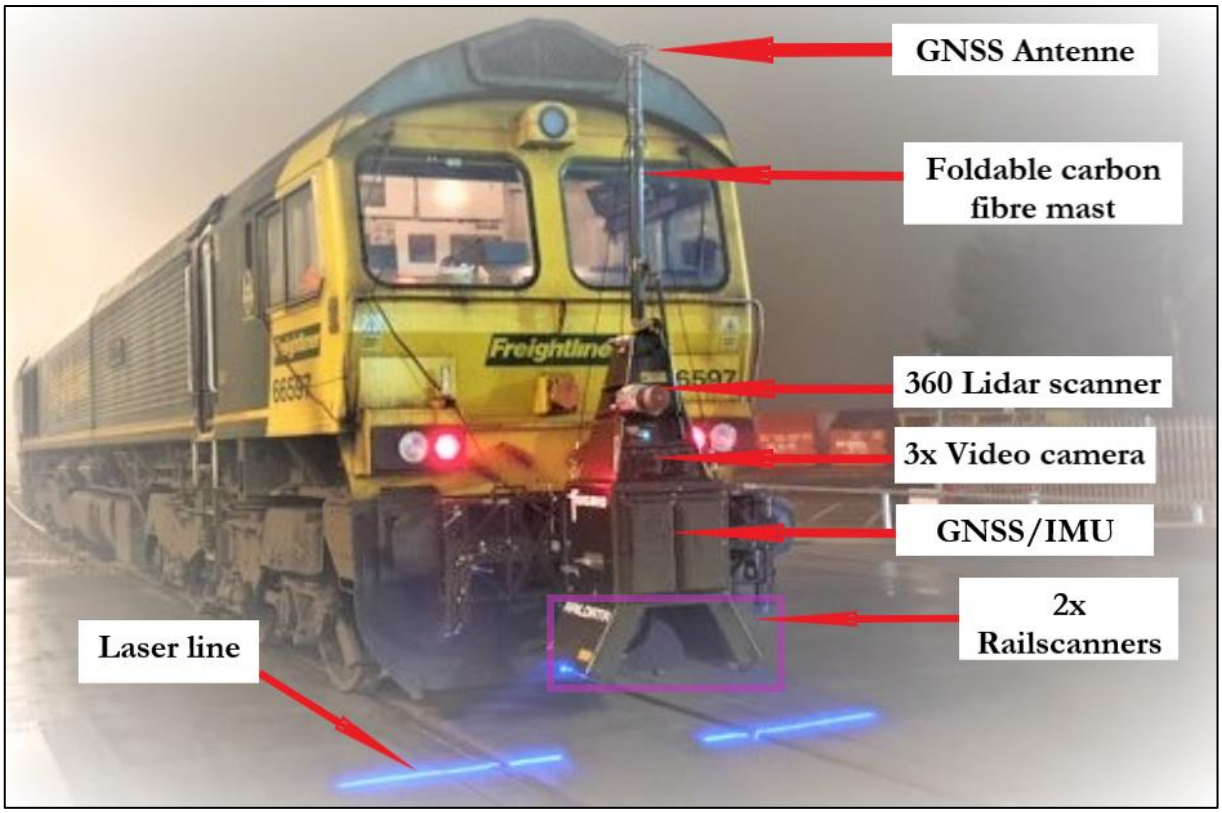


Figure 3-1: Fugro RailData's RILA system mounted on a train (source: Fugro RailData)

The data used for this research is for two simple switches located near the Soest station in the Netherland. Since the switch consists of two tracks, the data was acquired in two runs, whereby scanning is done per the rail being driven on. The stock and switch rails open on one side and close on the other in the first run, while the positions are interchanged in the second run. For this study, only data strips where the

stock switch rails are opened are considered. This is because the research aims to determine the flangeway distances, which is the minimum distance when the two rails are opened. Figure 3-2 depicts the left and right data strips for one of the switches where the stock switch rails are opened.

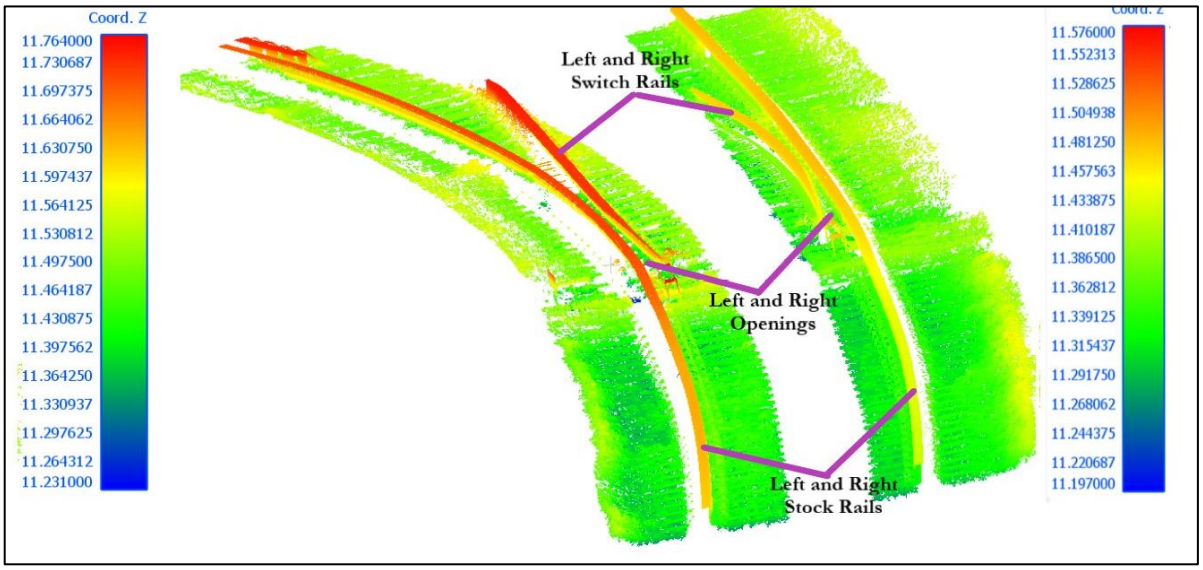


Figure 3-2: Strips of data for one switch indicating the opening on the right and left. The data has been coloured by height

Due to the scanning pattern during the data acquisition, the resolution of the points within the scanline on the rail is higher than those along the rail direction. The resolution on the sleepers is similar to that on the rail, while points on the ground have low resolution. Part of the rail web was occluded, resulting in no points in this area. Figure 3-3 has been coloured based on the point density to show the resolutions of the points as well as the occluded region. Although the point density on the railhead is high, there are sections of the rail where points are missing resulting in gaps or spares points. For some sections, the missing points resulted in the partial capturing of the rail head. An example of such areas is shown in Figure 3-4 below.

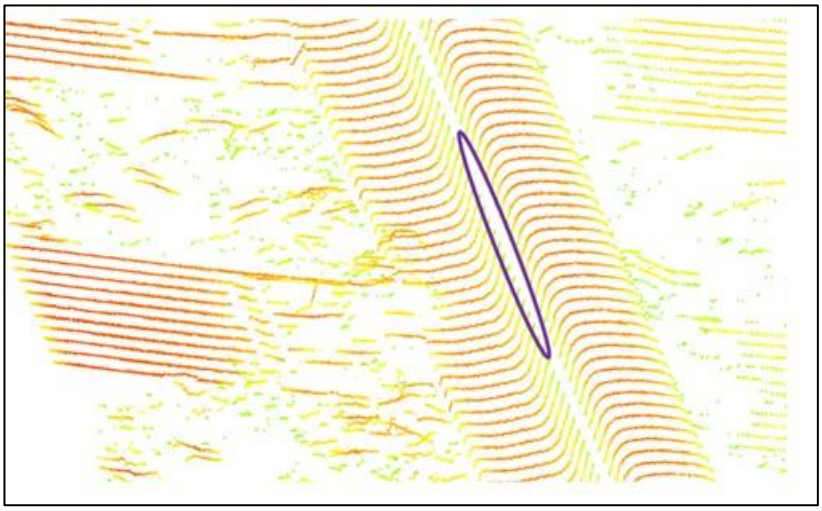


Figure 3-3: A Rail strip coloured by the point density; the violet ellipse indicates the no-data areas on the web

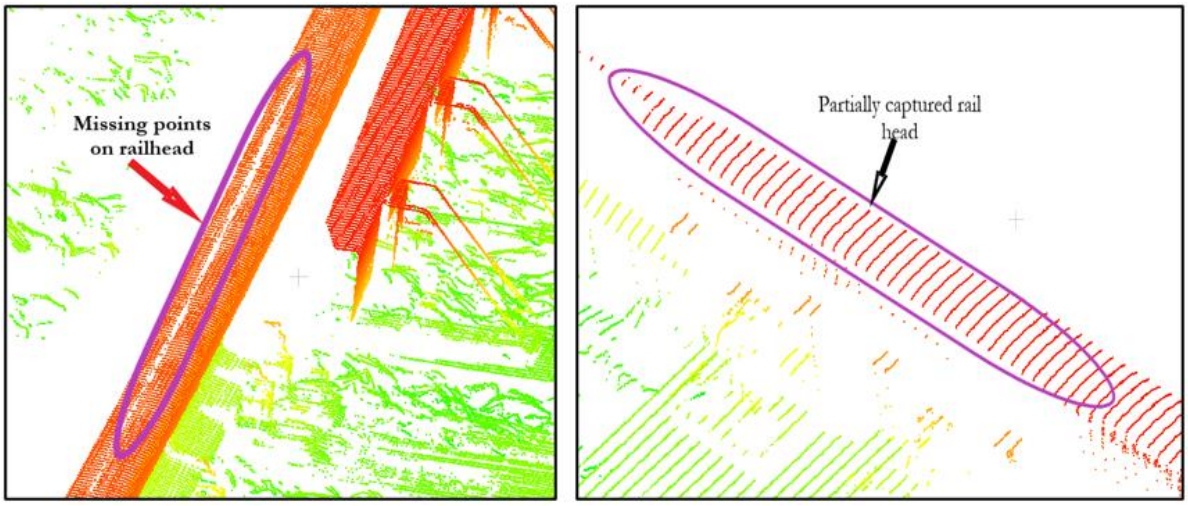


Figure 3-4: Left: Section of rail with missing points, which has resulted in gaps (marked with a violet ellipse). Right: Partial captured points on the railhead (marked with a violet ellipse)

The point clouds strips used in the research were acquired at two epochs (06/05/2020 and 03/09/2020) were used. A total of eight strips are used. Throughout this document, data strips are identified based on the date of acquisition, switch, and side of the track it lies. L and R are for strips on the left and right side; 1 and 2 are used to indicate that the strip is for the first and second switch, A and B indicate the first and second epoch, respectively. Table 3-1 and Table 3-2 gives further information on the point clouds strips used. All the strips used are shown in Appendix 1.

Table 3-1: Metadata of 3D point clouds

Measure	**Data Strips acquired on 06/05/2020**	**Data Strips acquired on 03/09/2020**
Points interval	Approximately 0.001 m on rail head and 0.002 m below the rail head	Approximately 0.002 m on rail head and 0.002 m below the rail head
Scanline interval	Approximately 0.02 m	Approximately 0.02 m
Data format	LAS	LAS
Number of data strips	Four (two per switch)	Four (two per switch)

Table 3-2: Information on data strips used

Date	**First switch/ID**	**Length (meters)**	**Number of points**	**Second switch/ID**	**Length (meters)**	**Number of points**
06/05/2020	L1A	20.45	4970307	L2A	22.82	3347296
	R1A	50.71	1703196	R2A	36.61	3693340
03/09/2020	L1B	20.19	1241213	L2B	22.63	3392515
	R1B	20.06	1241213	R2B	42.45	4072641

Additionally, because the data was acquired using an MMS, the trajectory data obtained with the point clouds during the data acquisition are used. The trajectory data contains the GNSS/IMU position and orientation angles (roll, pitch, and heading). The trajectory data is approximately between the right and left strips of the switch. Position and orientation angles (roll and heading) from the trajectory data are used for projecting the points when detecting the position of the running edge points for the rails.

3.2. Software used

Python programming language and Cloud Compare software are mainly used for this study. Cloud compare is used for data conversion, data visualisation, and connected component segmentation. All other methods are implemented in python. These methods are used for outlier removal, ground points removal, estimation of running edge, B-spline curve fitting, and distance computation to access the switch flangeway. Some of the libraries used apart from the normal python libraries are open 3D for outlier removal, SciPy for B-spline curve fitting, Sklearn for pair distance and minimum distance computation.

4. RESEARCH METHODOLOGY

4.1. Workflow

The proposed methodology for the research is shown in Figure 4-1. It begins with extracting separate subsets of points on the stock and switch rails from each strip (Table 3-2). The extraction of the rail points is achieved using segmentation. The segmentation of the rails is preceded by the removal of outliers and ground points. This is followed by determining the running edge positions for each rail using the segmented rail points. The running edges are the positions on the rail where the flangeway is measured. Due to the nature of data acquisition, there are no points in the inner face of the switch rail. Therefore, the running edge positions are determined at the outer face on the switch rail and the inner face on the stock rail. The algorithm employed for this stage uses the GNSS trajectory data from the RILA system together with the segmented rail points to determine the running edge positions. Then a continuous model is generated by fitting a B-spline curve to the running edge points for each stock rail and its corresponding switch rail. Distances are then computed between the stock rail and its switch rail models to determine the flangeway. The methodology of this study consists of three main stages. Figure 4-1 depicts the main workflow of methodology. Detailed descriptions of the processes and algorithms used are stated in various subsections of this study.

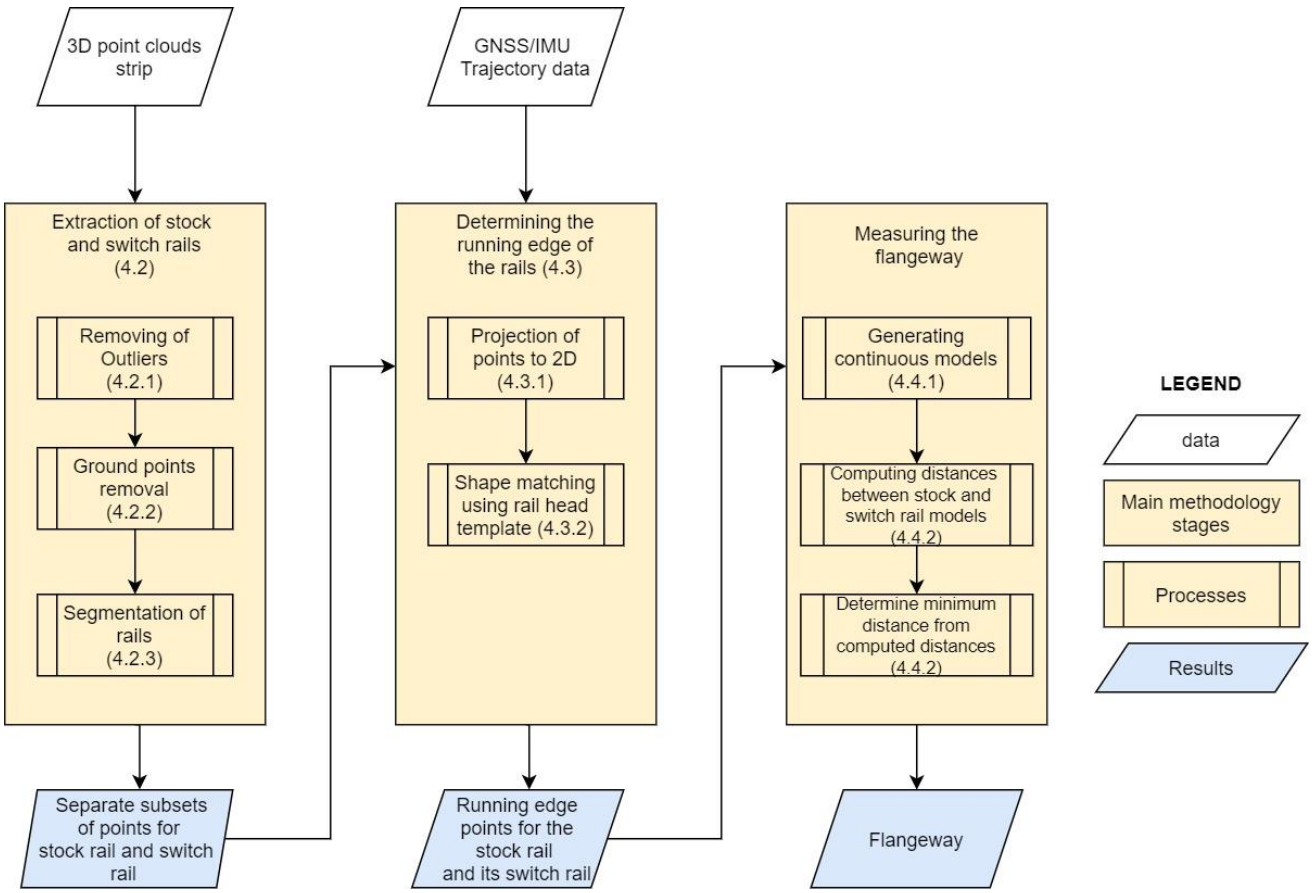


Figure 4-1: Flowchart of methodology workflow. Numbers in parenthesis represent the section numbers.

4.2. Extraction of stock and switch rails

The stock and switch rails are separate rails but are captured in a single strip of the point cloud. Therefore, they are extracted into separate subsets of points from each point cloud strip. Here, a connected component analysis is employed to segment the rails. Before the segmentation, outliers in the data and ground points that create connections between the two rails are removed. The outliers and ground points are removed for better segmentation of the rails. A statistical outlier remover and a grid-based elevation filtering algorithm are employed to remove the outliers and ground, respectively. Figure 4-2 below shows processes and the algorithms used to extract the stock and its switch rail from each point cloud strip.

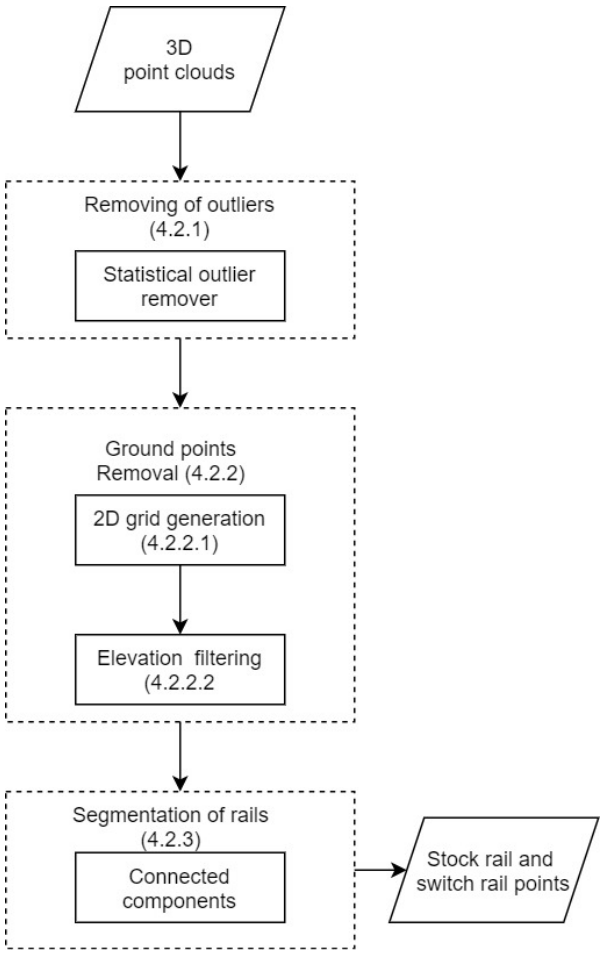
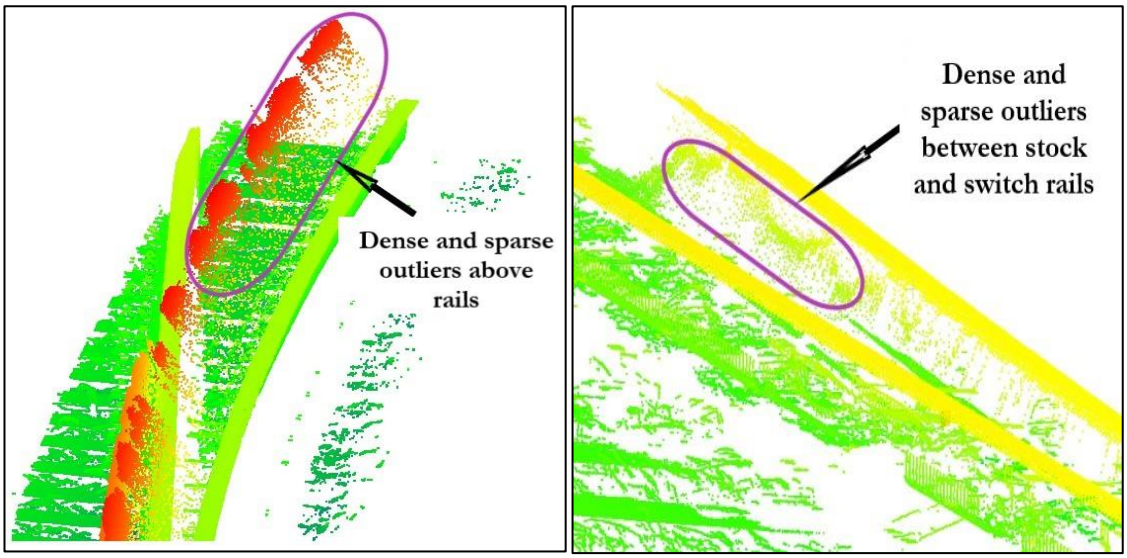


Figure 4-2: Workflow for section 4.2 showing the processes and algorithms used for extracting stock rail and switch rail

4.2.1. Removing of outliers

Outliers in point clouds are inevitable, e.g., as a result of reflectiveness from the scanning surface. The nature of the outliers can be sparse or dense. The dense clusters of outliers have high point density, while the sparse outliers have low point density and are either isolated or non-isolated (Ning, Li, Tian, & Wang, 2018). The non-isolated sparse outliers are often attached to the surface of the scanned object. The strips of the 3D point clouds used for this study also contain outliers. Some data strips have only sparse outliers, while others have both sparse and dense outliers. The outliers are either present above the rail or between the stock and switch rail. Figure 4-3 below depicts sections of the rail coloured by height to show the nature of outliers and their locations in the data. As shown in Figure 4-3 (a), some of the clusters of outliers have elevations higher than the rail points. The outliers in the data can influence the segmentation

and algorithm used to detect the rail's running edge, which will be discussed in later sections of this study. Therefore, these clusters of outliers are removed.



(a) (b)

Figure 4-3: Clusters of Outliers present at different sections of the data

A Statistical Outlier Remover (SOR) filter is applied to the strips to remove outliers in the data. The algorithm assumes that the distance between points in a specified neighbourhood is normally distributed. Therefore, the average distance of points in a specified neighbourhood is computed and assessed using the standard deviation. Points that are at a distance from other points and are not within the standard deviation of the mean distance are regarded as outliers and filtered out. The parameters for the algorithm are a user-defined neighbourhood threshold and standard deviation. The strength of the filter depends on the specified standard deviation. By using a high standard deviation value, few outliers are removed, while a low standard deviation value makes the filter stronger to removes more outliers. This means to remove sparse outliers; a high standard deviation value is required while a low standard deviation value is required for dense outliers.

The data strips used contained varying outliers with varying densities; using a smaller standard deviation resulted in removing points around the rail head. These points are removed in areas where there are data gaps around the railhead and around the section closer to the upper rail web. The SOR filter removes points in these sections because they are quite sparse. Hence, several experiments are conducted to determine the set of parameters that could remove the outliers as well as preserve the points on the rail head. The parameters in Table 4-1 are then chosen based on experimentation. However, it should be noted that the resulting point clouds for some of the strips are not completely outlier free. This is because the standard deviation used is high, making the algorithm less aggressive to remove some of the outliers with a distribution similar to that of the rail head points.

Table 4-1: Parameters for statistical outlier remover

Parameters	**Value Used**
Neighbourhood size	100
Standard Deviation (m)	0.9

4.2.2. Ground points removal

The ground points removal also includes removing points on track bed objects such as sleepers and rail supports. Points on the rail web are also removed because we are only interested in points on the rail head. The motivation for this step is to make further processing faster and remove the points that create connections between the stock and switch rail for better segmentation. As the rail head points are of interest here, a similar approach used by Oude Elberink et al. (2013) for rough classification of rails is employed. The approach first analyses the height distributions of points in a specified grid cell and then selects rail points based on the elevation of points in the grid. Due to the slope of the terrain, which results in varying elevation of the rail points, using the grid helps to inspect the points and also performs the filtering of rail points locally. Section 4.2.2.1 and 4.2.2.2 gives a detailed explanation of the approach used.

4.2.2.1. 2D grid generation

Generating the 2D grid begins with creating an oriented bounding box for the denoised strips from section 4.2.2 using the OrientedBoundingBox algorithm provided by the Open 3D library. This bounding box has its axes aligned to the principal axis of the data and therefore fits well to the data. Since the data strip is a one-sided rail that is somewhat thin in dimension, using the oriented bounding box for generating the 2D grid prevents empty grid cells. The oriented bounding box created for one of the data strips is shown in Figure 4-4.

The 2D grid is then created by determining the range of the grid in the x and y directions. This is achieved by finding the x and y coordinates of the oriented bounding box. The range of the x-axis is determined using the difference between the minimum x and maximum x, and a similar calculation is done to determine the range of the y axis. These ranges denote the width and length of the bounding box, respectively (Equations 4-1 and 4-2 below). Based on the computed range, the cells in the grid are defined by dividing the range in the x-axis into smaller segments of 0.5 m and the y-axis by 2 m. The results are rectangular grid cells of a size of 0.5 m for the x-axis and half the bounding box's width for the y-axis. The number of rows and columns in the grid is determined by Equations 4-3 and 4-4 below. The chosen grid size ensures that most of the cells will have rail points.

$$lenght\ of\ bounding\ box = (maximum\ x) - (minimum\ x) \quad \text{Equation 4-1}$$

$$width\ of\ bounding\ Box = (maximum\ y) - (minimum\ y) \quad \text{Equation 4-2}$$

$$number\ of\ rows = \frac{length\ of\ bounding\ box}{0.5\ m} \quad \text{Equation 4-3}$$

$$number\ of\ rows = \frac{width\ of\ bounding\ box}{2\ m} \quad \text{Equation 4-4}$$

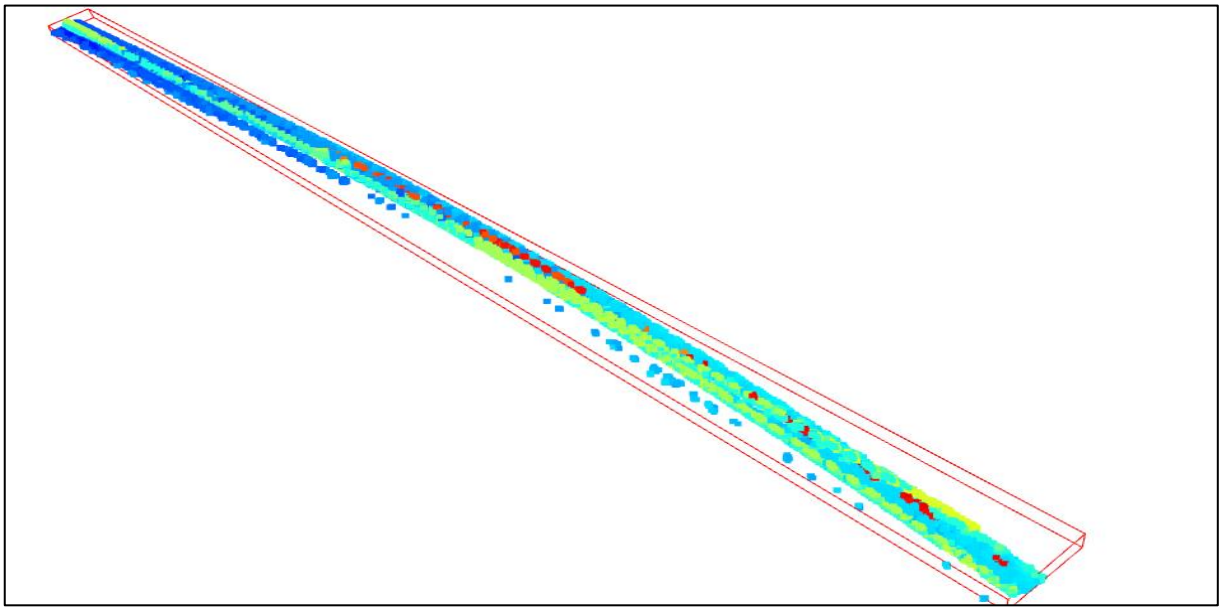


Figure 4-4: Oriented bounding box is shown in red around a data strip

4.2.2.2. Elevation Filtering

The idea of the elevation-based filtering algorithm is based on the fact that the rail is higher than other objects on the track bed. The key here is to determine the elevation of points on top of the rail, which are used together with the known height of the rail head to obtain points belonging to the rail in each cell. The algorithm is implemented by first selecting points within each rectangular grid cell which has coordinates greater than the minimum and less than the maximum x and y coordinates of the cell. An example of points selected by one of the grid cells is depicted in Figure 4-5.

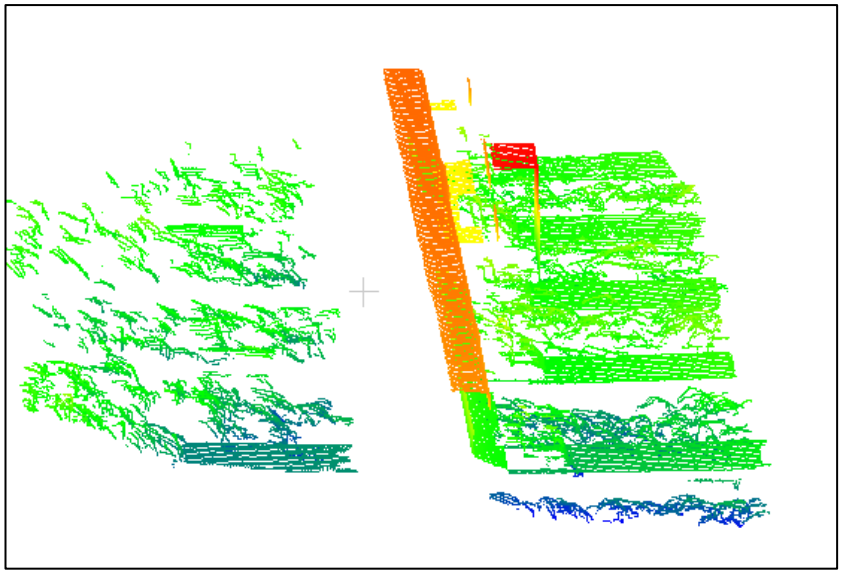


Figure 4-5: Points selected by a grid cell coloured by height. The cell size is defined as 0.5m for the x-axis and half the width of the bounding box for the y-axis

For the selected points in each cell, a point on top of the rail is determined, and its elevation is used as the base elevation for filtering. There are other objects in the data as well as some cluster of outliers above the rail after the outlier removal; therefore, the point on top of the rail is determined by analysing the height distribution of points in each cell. This is done by sorting the points in a grid cell based on their elevation and finding the point at a certain percentile height. The point on top of the rail per cell is determined at the 90^{th} percentile for data strips with outliers above the rail and the 98^{th} percentile for strips with no outliers. Knowing the height of the rail head, which is approximately 0.05 m, a threshold is defined by computing the difference between the elevation of the percentile points determined and rail head height. Points whose elevation are higher than the defined threshold are then considered rail points, and the other points on the track bed and ground are removed.

4.2.3. Segmentation of rails

The motivation for this step is to separate both rails into individual subsets of points. Since the two rails appear as continuous objects with a space between them, connected component analysis is employed to obtain the individual rail points. The connected components is a region-based growing method but do not require seed point when segmenting point clouds (Chen et al., 2019). With the algorithm, the relationship between points within a certain distance in a neighbourhood is investigated, and points with similar properties are grouped into one segment (Che, Jung, & Olsen, 2019).

The connected components algorithm requires a grid step size which is the smallest distance between two components to be defined. Since the method is implemented in the cloud compare software, the grid step is determined at an octree level based on the dimension of the data. Therefore, the algorithm requires two parameters: the octree level and minimum points per segment to be defines. The data is divided into eight 3D grid cells at an octree level, which is further divided into more cells when the level is increased. Then connected points within a grid are grouped to form one component. The parameter for minimum points per segment helps to remove small components.

The algorithm's implementation begins with a trial-and-error approach to determine the combination of parameters that results in a good segmentation of the rails. It is observed that the rails are well-segmented at a grid step of approximately 0.01 m. For all the rails except one (L1A), this grid step is obtained at an octree level of 11. Strip L1A is longer than all the other strips; therefore, the required grid step (approximately 0.01 m) is obtained at an octree level of 12. The component numbers generated after segmentation are then used to select the stock and switch rails points manually from the remaining components using visual assessment.

4.3. Determining the running edge of the rails

This stage takes the individual stock and switch rail points obtained from section 4.2.3 as input. Determining the rails' running edge is to obtain the positions where the flangeway is measured. As stated earlier in section 4.1, since there are no data in the inner face of the switch rail, the running edge position for the switch rail is determined at the outer face. The running edge position is known from the designed rail profile (Figure 4-8 a). Therefore, the idea of this stage is based on a brute force matching approach whereby the rail template is matched to the points to determine the positions of the running edge. The trajectory data, which contains the GNSS/IMU positions and orientational angles, is also used. The trajectory data lies parallel to the rail, and it is approximately similar to the centre line of the tracks. This stage is carried by traversing along the trajectory data at an interval of 0.1 m. At this interval, a buffer is defined to select points on the rail; these points are projected to 2D. The rail head template is fitted to the

projected points to estimate the running edge positions on the rail. Since there are missing data and gaps on the rail head, this approach is adopted so that the point density within a scanline can be increased to enable the best fitting of the rail template. Figure 4-6 depicts the steps used in estimating the running edge positions for each stock and switch rail. The steps are further elaborated in sections 4.3.1 and 4.3.2.

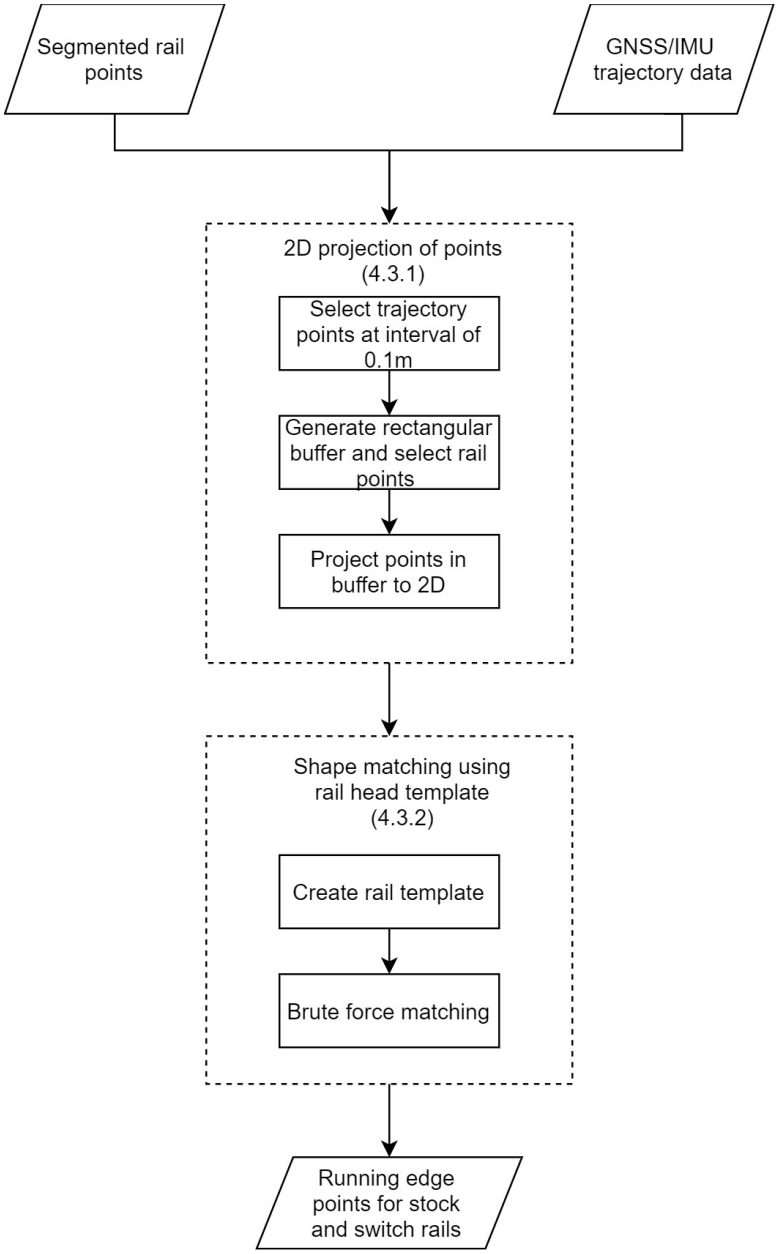


Figure 4-6: Workflow for section 4.3 showing the processes and algorithms used for determining the running edge points

4.3.1. 2D projection of points.

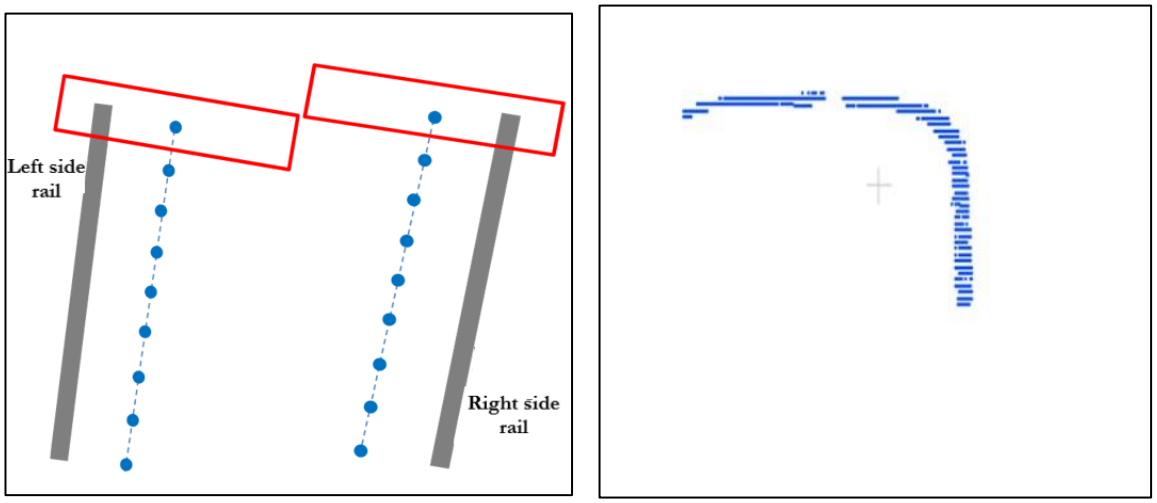
This stage begins with selecting points at a defined interval from the trajectory data. This is because the running edge points are determined at the trajectory point position along the rail. The trajectory data for each strip is different. Points in each of these trajectory data also have different intervals. However, we want to determine the running edge positions at the same intervals for each strip. Therefore, an interval of 0.1 m is defined to select points in the trajectory data. The points are selected by computing the distance from the first point to subsequent points in the trajectory data. The next point away from the first point whose distance is equal or greater than the defined interval is selected. This step is done iteratively to obtain trajectory data with points of interval 0.1 m. These new trajectory points are then used for further processes in this stage. An interval of 0.1 m is used because it is expected that the rail is approximately linear at this interval, and its direction does not change.

For every point location in the trajectory data obtained above, a rectangular buffer similar to the one depicted in red in Figure 4-7(a) is generated. Based on knowledge of the track's gauge distance and the width of the rails, the length of the buffer is defined as 3 m. At this length, the buffer is generalised so that rails on either side can lie inside to be selected. Therefore, the buffer is defined such that half of its length lies on the left and right sides from the trajectory point. The width of the buffer is also defined to be 0.1 m. The width controls the point density in the buffer, and it is important for fast and accurate processing. A larger width selects more rail profiles, resulting in a higher point density and increasing the processing time. A smaller width also selects few points, which may also affect the accuracy of the estimation. Here also, half the width is defined to be below and above the trajectory point. Figure 4-7 (a) depicts how the rectangular buffer selects rail points on both the left and right sides. Points on the rail which fall within the buffer are selected using the bounding coordinates of the buffer.

The selected points are projected to a vertical cross-sectional plane that is perpendicular to the trajectory's direction. The plane's rotation and translation are defined using the yaw angle and coordinates of the trajectory point at the time of acquisition, respectively. The coordinates of the points on the plane are obtained using Equations 4-5 and 4-6 below where (x_t, y_t) and θ are the coordinates and yaw angle. Figure 4-7 (b) depicts the projected rail points which are selected in the buffer at a trajectory point location.

$$x = [(x_p - x_t)\cos\theta] - [(y_p - y_t)\sin\theta] \quad \text{Equation 4-5}$$

$$y = [(x_p - x_t)\,sin\,\theta] + [(y_p - y_t)\,cos\,\theta] \quad \text{Equation 4-6}$$



(a) (b)

Figure 4-7: Image depicting how the rails on the left and right are selected using the rectangular buffer. The rectangular box is shown in red; the rail is shown in grey; trajectory data is shown in blue with the resampled points as blue dots. (a) The rectangular buffer when the rail is on the left side and right side (c) 2D projected points with a width of about 0. 1 m.

4.3.2. Shape matching using rail head template

The idea here is to obtain a match between the railhead template and the projected 2D points. The running edge points are obtained when the best fit is achieved. Therefore, a brute force matching approach is used to obtain the best fit between the 2D points and the railhead template. The general idea of the algorithm is to compare two shapes to obtain the best position where the shapes match each other. The best position is obtained when the match between the features in the first shape to features in the second shape returns a minimum distance. The rail head template and 2D projected points obtain the best match when aligned at the running edge. This means that the distance between the rail head and the points should be minimum at this location. For each 2D point location, a mean distance is computed between the railhead template and the 2D points in the window. The point location that returns the minimum mean distance is chosen as the running edge point.

The implementation of this approach begins with creating the rail template using the rail head dimensions. Since part of the rail was occluded during the data capture, the railhead template is created using three-line segments based on the half dimension of the railhead. The dimension of the railhead used to create the template is shown in Figure 4-8(a). This dimension corresponds to only the top and side of the rail head template. The template is created based on the UIC54 rail profile Figure 4-8 (a and b), which is the common profile used by the data providers for rail detection. The centroid of the template is defined at the running edge location of the railhead.

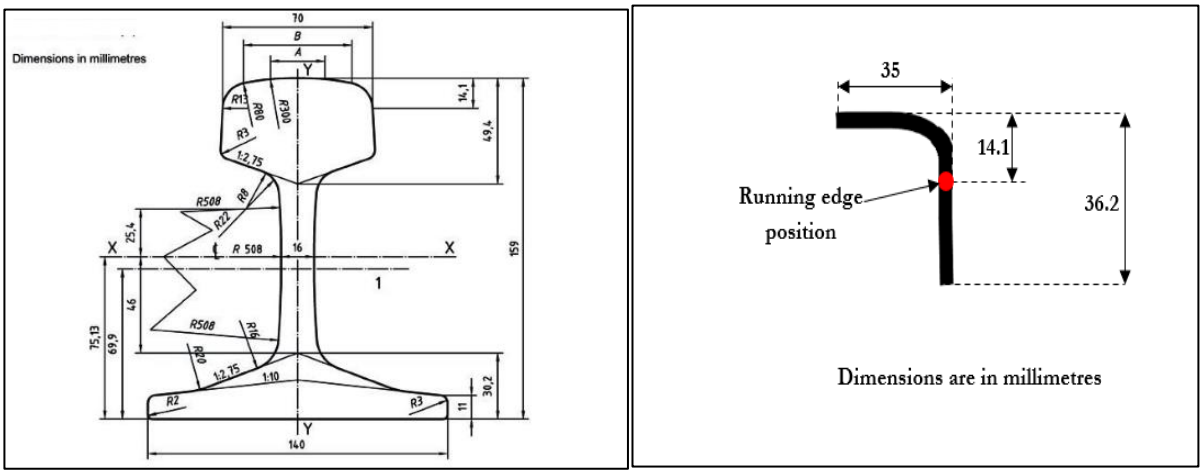


Figure 4-8: (a) UIC54 rail profile (b) dimension of rail head used to design template

The rail head template created is projected to each 2D point location to determine the location for the best match. The projection is achieved using the equations 4-5 and 4-6 above for the projection of the 2D points. However, the rotation and translation are defined by the roll angle and coordinates of the trajectory point. When the template is projected to a point location, two buffers are defined. The first is generated with select only rail points to be matched. This buffer eliminates the remaining points attached to the rail head, which are not removed after segmentation. The second buffer is generated using the coordinates of points at the template's start and end line segment. This buffer further reduces the number of points to consider in calculating the minimum distance and makes the algorithm faster

After, the minimum distances between points within the second generated buffer and the horizontal and vertical line segment of the template is computed. The mean of the minimum distances for both horizontal and vertical line is then computed to represent the distance between the railhead and the projected points. The best match is achieved at the point location when this mean distance is minimum. The minimum number of points to consider for the minimum mean distance is five (5). This helps to eliminate false match and determine the running edge on the tip of the switch rail. Also, a distance of 0.015 m for the horizontal and 0.03 m for the vertical line segments is defined as the deviation between the points and template. This step was done iteratively. Finally, the point location where the best fit is achieved is selected as the running edge at that trajectory point location. The points are projected back to 3D.

The method described in this section is applied repeatedly for every 0.1m trajectory point location. It is applied similarly for all the stock and switch rails extracted for both the left and right side. For rails on the left side, the template as described above is used for matching, and this template is mirrored to match the rails on the right. It should be noted that the switch rail has a different profile from the stock rail. It is made of a full rail similar to the stock but differs in width; this section transitions into a knife-edge shape, getting to the tip of the switch rail. However, since the switch rail profile was not known at the time of conducting this study, UIC 51, the stock rail profile, is used. Furthermore, after the running edge points are determined, they are stored in a text file and visualised Cloud Compare software. The points for both rails in the region of interest are then selected based on visual assessment. The region of interest is from the tip of the switch rail to the area where the gap between the two rails widens.

4.4. Measuring the flangeway

This stage takes the selected running edge points for the region of interest from 4.3.2 as input. A continuous rail model for each stock rail and its corresponding switch rail is obtained by fitting a B-spline curve to the points to generate the continuous models. Distances are then computed between the stock rail and switch rail models. From the computed distances, the flangeway is determined. Figure 4-9 is the step-by-step approach used to achieve the flangeway for each strip.

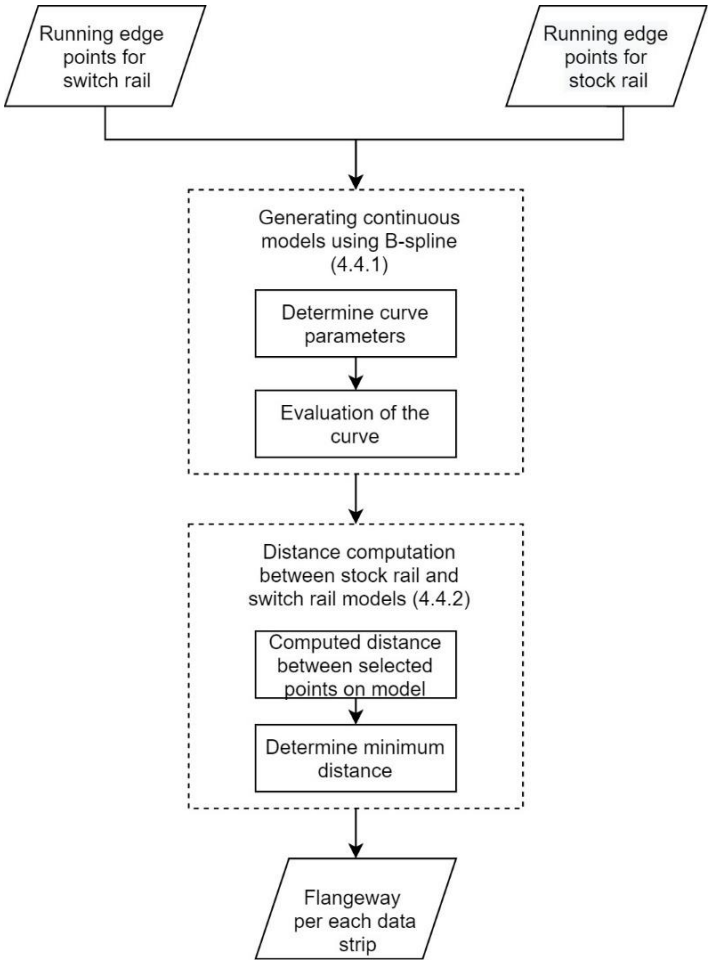


Figure 4-9: Workflow for section 4.4

4.4.1. Continuous rail models with B-spline curve fitting

Generating a continuous model allows for continuous distance measurement, even when the curvatures of the stock rail and switch rail changes. The B-spline is a linear combination of basis functions and control points given by Equation 4-7 below (Harmening, 2020).

$$S(x) = \sum_{I=0}^{np} N_{i,p}(u).P_i \qquad \text{Equation 4-7}$$

Where $N_{i,p}(u)$ is the basis function and P_i is the control point.

The basic functions are piecewise polynomial, which are determined by the degree of the curve and knot vectors (Farin, 2002). The knot vectors are the points where the piecewise polynomial meet and are made up of non-decreasing knots. Two steps are used to fit the curve to the points. The first step involves determining the parameters that represent the curve. These are the knot vectors, degree of the curve, smoothening factor and the location parameter, which determines the location of each data point on the curve. For this study, the degree of the curve is defined as three (3) and the smoothening factor as s $m - \sqrt{2m}$ where m represent the number of data points to fit the curve. These parameters are determined based on the recommendation given in the documentation of the SciPy library[1] used. With these predetermined parameters and the x and y coordinates of the running edge points, the knot vector and the location parameters were also determined. The curve representation parameters are used to evaluate a number of control points. The control points determine the shape of the curve. The number of points evaluated for each stock rail and switch rail depended on the length of the running edge points used. The results from this step are continuous models for the stock and switch rail, whereby each model consist of a dense set of points.

4.4.2. Distance computation between stock and switch rails models

The dense set of points generated for each model in section 4.4.1 are used to compute the distances. For clarity, the dense points generated for the switch rail is referred to as set A and that of the stock rail as set B. For each point in set A, the closest point in set B is found. Euclidean distance is computed between each point in set A and their closest points in set B. The results obtained are the distances between the stock rail and switch rail. From these computed distances, the minimum distance, which is the flangeway is determined.

[1]https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.splrep.html#scipy.interpolate.splrep

5. RESULTS

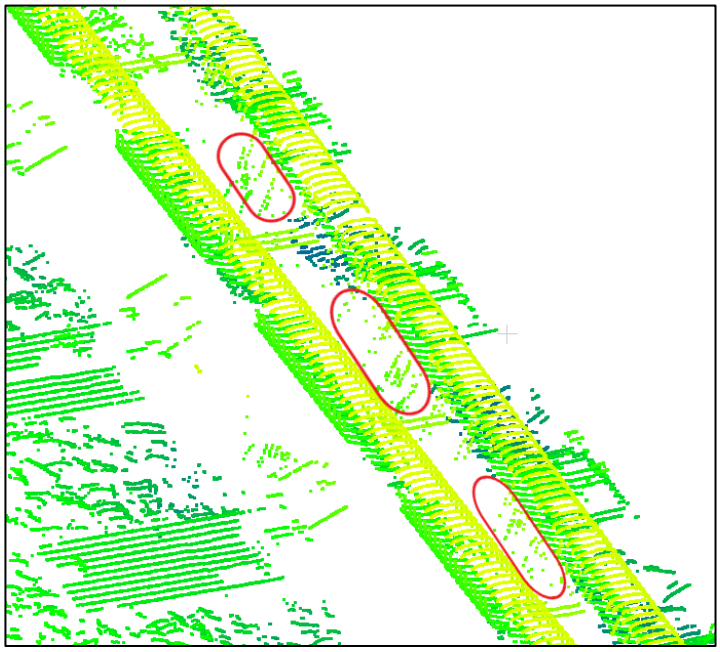
5.1. Extraction of stock rail and switch rail

Extraction of the stock and switch rails begins with removing outliers present in each data strip using a SOR filter. The parameters used are a neighbouring size of 100 and a standard deviation of 0.9 m. The results obtained reveal that for strips that had no data gaps around the rail head and few sparse outliers, the stated parameters removed the outliers present while preserving the rail head points. Because the results for these strips are similar, one is shown in Figure 5-1 below. Figure 5-1 (a) and (b) shows the strip before and after the statistical outlier remover has been used to remove the outliers, respectively. By comparing the two images, it is observed that the outliers indicated in red ovals between the stock rail and switch rail in Figure 5-1 (a) has been removed in Figure 5-1 (b). Some of the strips had gaps around the rail head and few sparse outliers. For such strips, the results are similar to those presented in the scenario above.

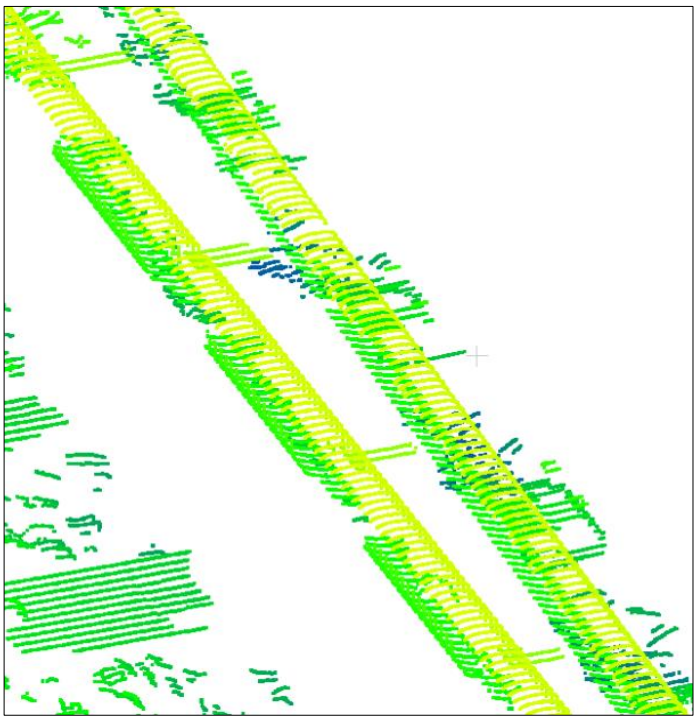
Another category of strips had gaps around the rail head and varying clusters of outliers at different sections of the rail. For strips in this category, some of the sparse outliers in the data are removed except for those attached to the rail surfaces. Also, some of the dense clusters of outliers present above the rail are not removed by the SOR filter. This is as a result of the large standard deviation value used to preserve the rail head points. It was observed that the clusters of outliers that are not removed have a similar distribution to the point on the rail head points. Therefore, the algorithm did not consider them as outliers. An example of the result for strip in this category is shown in Figure 5-2. Figure 5-2 (a) shows the data strip before the outliers are removed, while Figure 5-2 (b) shows after the outliers are removed. By comparing the two images, it is observed that some of the sparse and dense cluster of outliers indicated in the purple oval in Figure 5-2 (b) are not removed. The images are coloured by height; therefore, the dense outliers are coloured in orange in the purple ellipse because their elevations are higher than the rail, while the sparse outliers on the surface have the same colour as the rail due to similar elevation with the rail points. Table 5-1 shows the category for each data strip according to the observations made.

Table 5-1: Data strips for each category after outlier remover

Strips with no gaps and few sparse outliers	Strips with gaps and few sparse outliers	Strips with gaps and varying clusters of outliers
L1A	R1A	R1B
L1B	R2B	R2A
L2A	L2B	

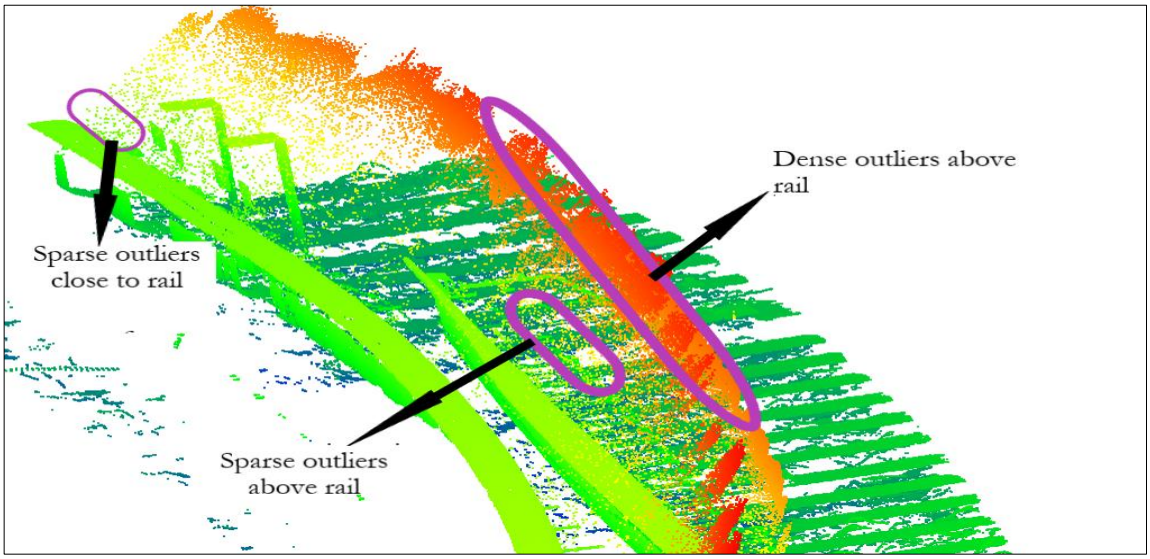


(a)

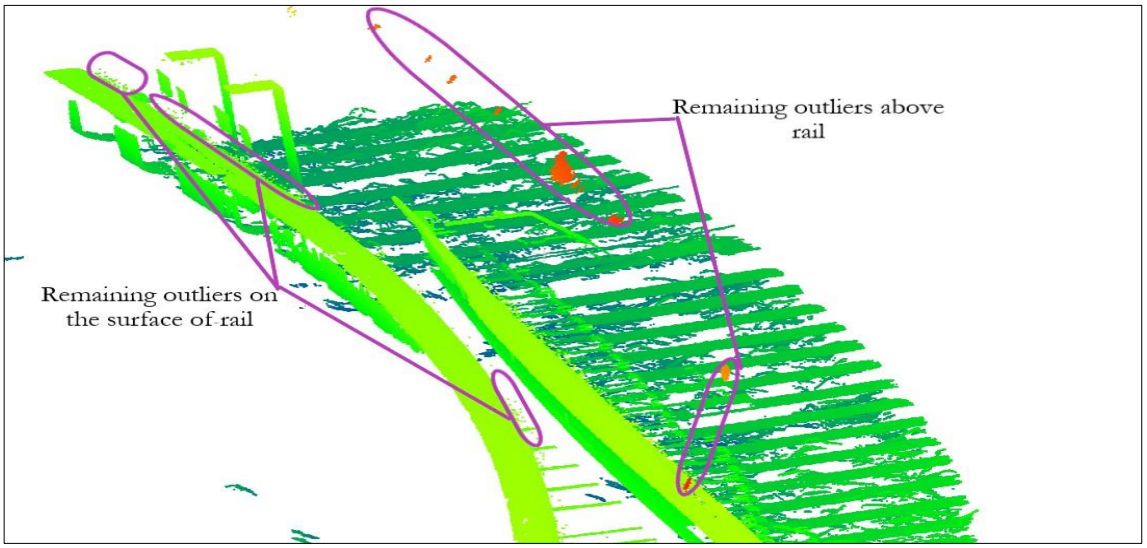


(b)

Figure 5-1: Section of the data strip coloured based on height to show sample results for strips with very sparse outliers and no gap on rail head. (a) Depicts the data strip before removing the outliers; the red ellipse indicates the outliers between the stock rail and switchblade. (b) Depicts the same data strip after the outliers have been removed.



(a)



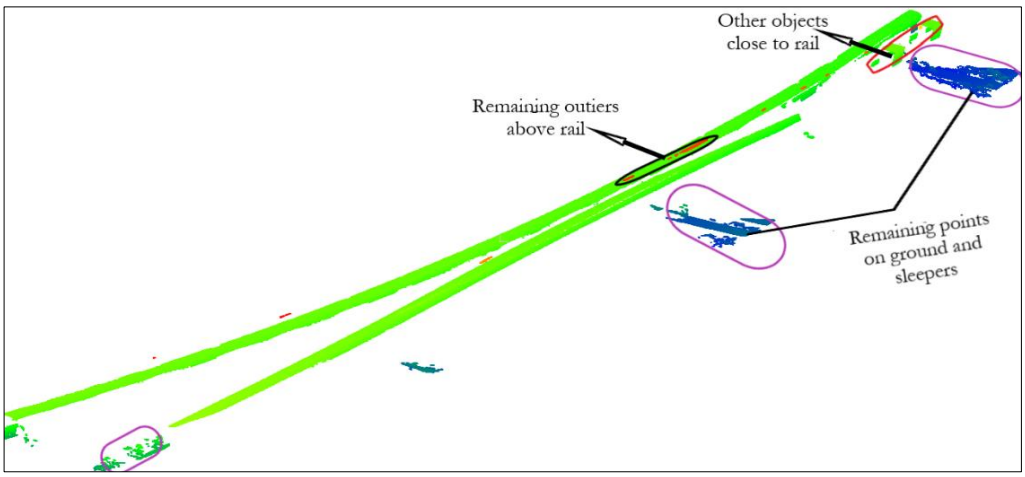
(b)

Figure 5-2: Section of the data strip coloured based on height to show sample results for strips with both sparse and dense outliers and gaps on rail head. (a) Depicts the data strip before removing the outliers; the red ellipse indicates the outliers between the stock rail and switchblade. (b) Depicts the same data strip after the outliers have been removed.

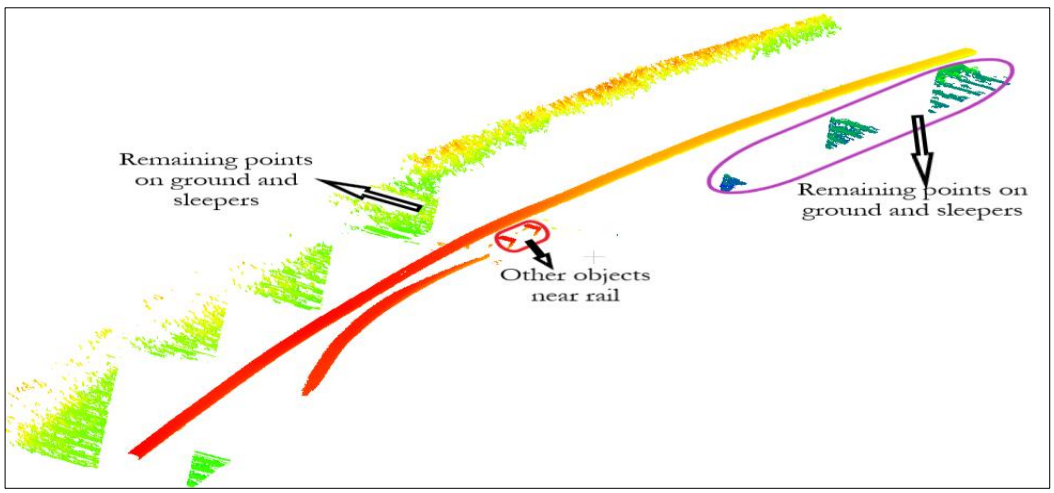
The next step after the outlier removal is the removal of points on the ground. The points on the ground are removed based on the elevations of points in a grid cell. The distribution of points in each cell is considered to get rail head points. The elevation of the rail head point obtained at a percentile height in each cell is then used together with the height of the railhead to filter out other points on the ground. Therefore, the point at the 98th percentile is considered for strips with no remaining outliers above, while the point at the 90th percentile is considered for data strips with remaining outliers above. There is only one strip with outliers above (R1B). Therefore, the 90th percentile is used for only one strip, while the 98th percentile is used for the remaining data strips from both epochs. Figure 5-3 shows examples of results obtained after filtering out points on the ground. From the images, it is observed that ground points in between the rails are removed. However, there are still some non-rail points that are not removed. These are outliers, rail support, and some ground/sleeper points outside the location of the rails. The outliers are not removed because their elevations are higher than the threshold used for selecting the

rail points. The rail support points are also not removed because their elevations are similar or slightly higher than the railhead points.

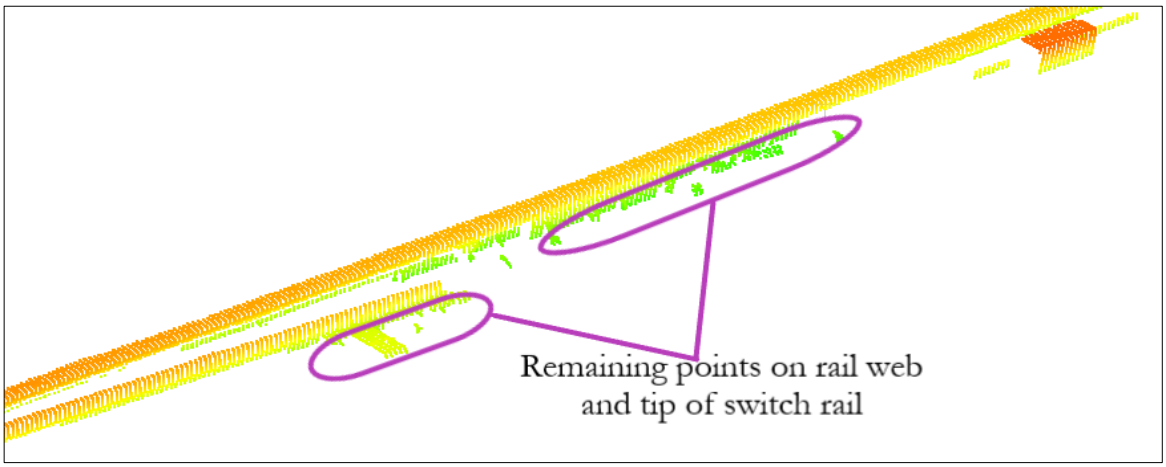
Furthermore, it is observed that not all the grid cells had rails in them. Since the filtering is performed locally per grid cell, the height percentile points in these cells were not rail head points; therefore, points in such grids are not removed. This resulted in the remaining points on the sleepers and the ground. Comparing the images in s 5-3 (a) and (b) shows that the remaining points on sleepers and ground are more in the former than the latter. This is because the original strip in the first image has fewer terrain points than in the second image. The remaining non-rail points after the elevation filtering had no connections with the rail; therefore, no further attempt is made to remove them during this stage. This is because they would easily be segmented out after the connected component segmentation. Again, it is observed that some cells had multiple rails, and in some cases, the height percentile point in the cell is located on the switch rail. In such grid cells, some points below the rail head also remained. These points are mainly on the upper rail web or at the tip of the switch rail and are connected to the rail head. An example of the result for such instance is shown in Figure 5-3 (c). This happened due to the slight variation in the elevations between points on both the stock and switch rail.



(a)



(b)



(c)

Figure 5-3: Results after elevation filtering. (a) A data strip with outliers above the rail with fewer remaining terrain points. (b) A data strip without outliers above the rail with more remaining terrain points. (c) A data strip showing the remaining points on the web and tip of the switch rail.

Connected components algorithm is applied to obtain the individual stock rail and switch rail points. The results obtained are shown in Figure 5-4 and Figure *5-5* below. The components generated after the segmentation are shown in Figure 5-4. The components are depicted with colours; points in the same component have the same colour. The image shows that the stock rail and switchblade points are shown with different colours, which implies that the components are different. Also, the remaining ground points and the outliers above the rail from the previous method formed separate components. However, for strips where the remaining points are connected to the rail head, these points are included in the rail's component (Figure 5 5). In some cases, the switch rail is slightly over segmented at the region close to the tip because there was a gap in this section (Figure 5-5). The components for the stock rail and that of the switch rail are selected manually based on the component numbers. It was observed that the first component in every segmented strip was for the stock rail, while the second was for the switch rail. This is because they contained the largest number of connected points. In cases where over-segmentation occurred at the tip of the switch rail, the smaller components are also merged manually to the main components. This is also shown in Figure 5-5. In the image, the original colours created for the components are used to indicate each rail, as well as the over-segmented area shown as grey on the switch rail.

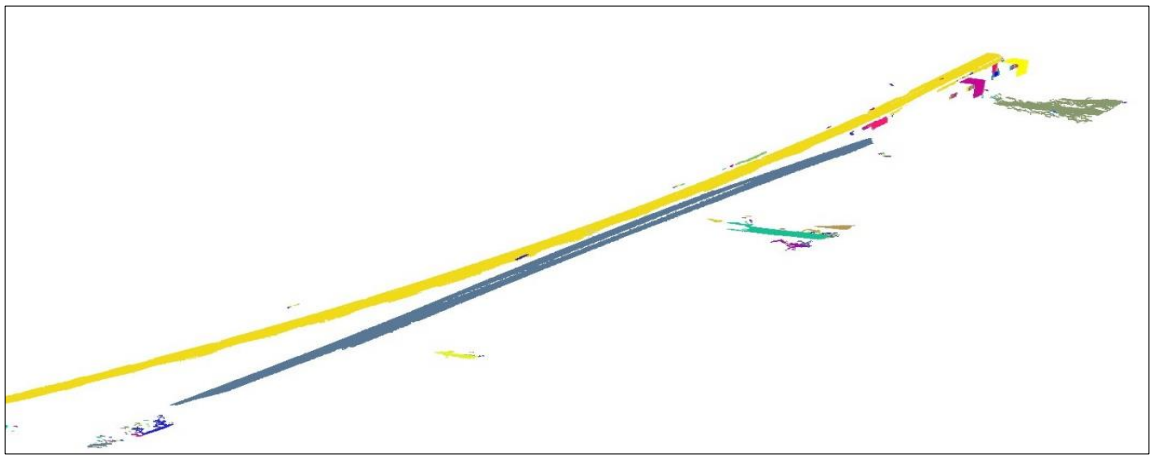


Figure 5-4: Results showing the components created after segmentation of the rail. Points in the same component are indicated with the same colour.

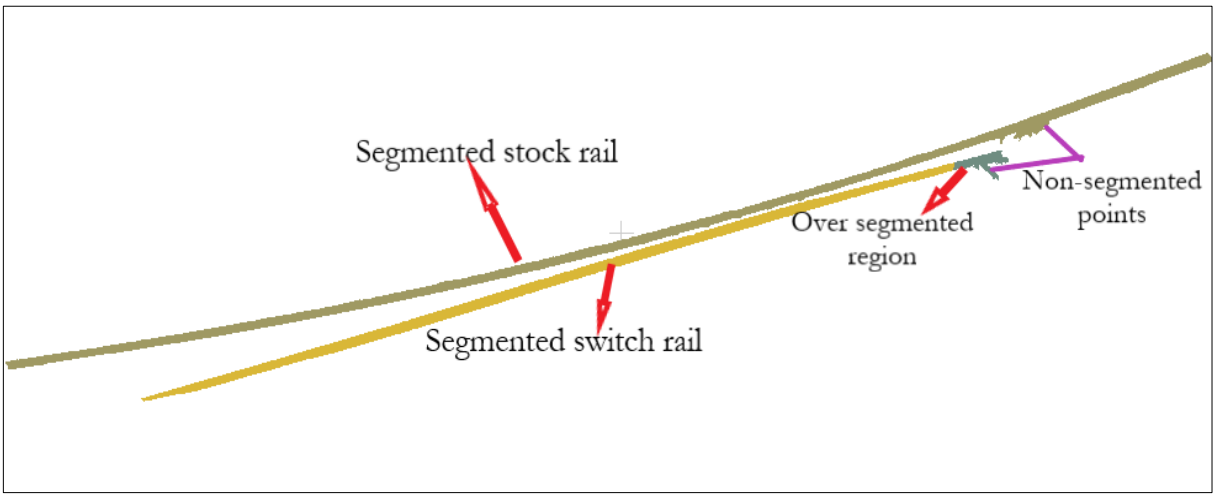
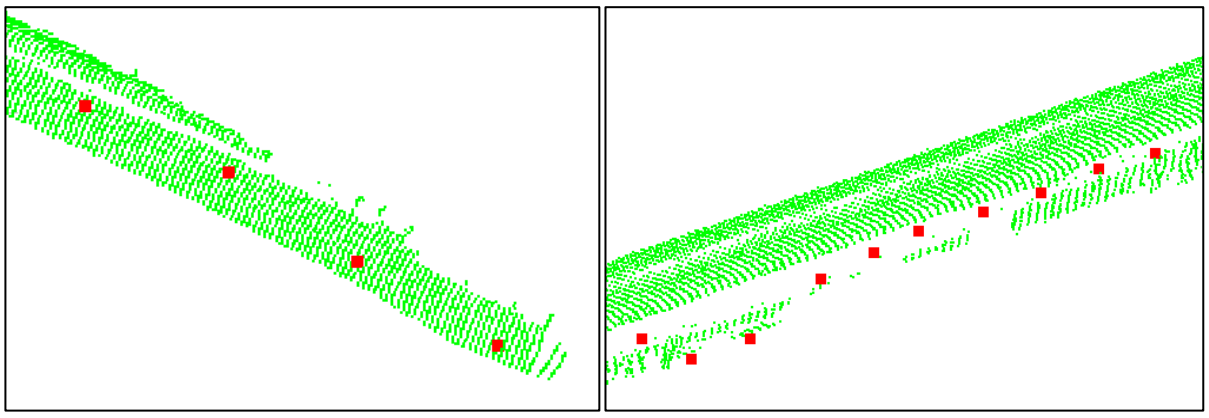


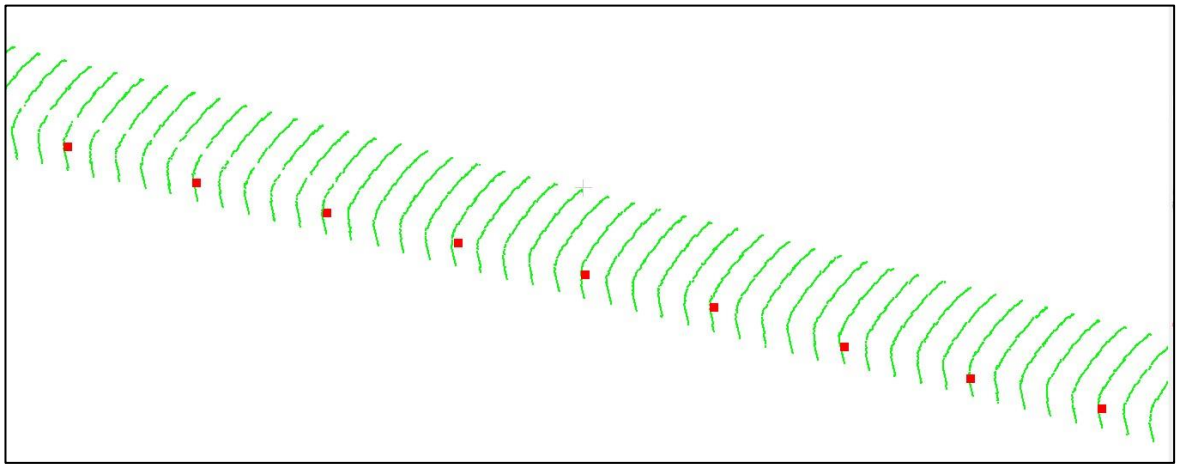
Figure 5-5: Stock and switch rail selected as individual rails after segmentation. Also, the over-segmented region and the non-segmented points are shown.

5.2. Determining the running edge of the rails

The Running edge along the individual stock and switch rails obtained from the segmentation are determined and saved as 3D points. As the profile used for the switch rail is quite different from the actual profile, the determined running edge points are expected to be different from the actual position. This deviation is in the section close to the tip of the switch rail (Figure 5-6 (a)). Visual assessment reveals that the running edge positions for most of the stock rails and other sections of the switch rail are accurately determined. However, the determined points deviated from the expected position for areas where the rail head had many missing points. Figure 5-6 (b) shows an example of running edge points determined for such rail sections. Figure 5-6 (c) shows a section on the rail where the running edge points are determined at the expected position. The determined running edge points in the region of interest are selected based on visual assessment. An example result of the selected running edge points for a pair of stock and switch rail is shown in Figure 5-7.



(a) (b)



(c)

Figure 5-6 Limitations observed after determining the running positions of the rails. (a) Limitation relating to the switch rail. (b) Limitation relating to missing points on rail head. In both figures, the rail is shown in green and the running edge points in red. (c) section of the rail showing the running edge at the expected position.

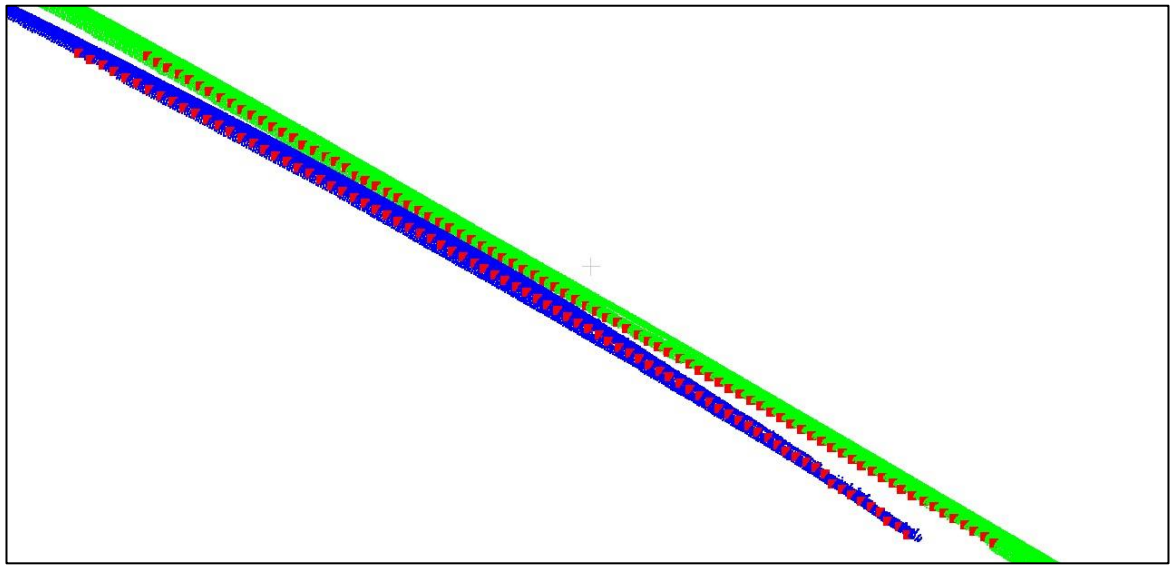
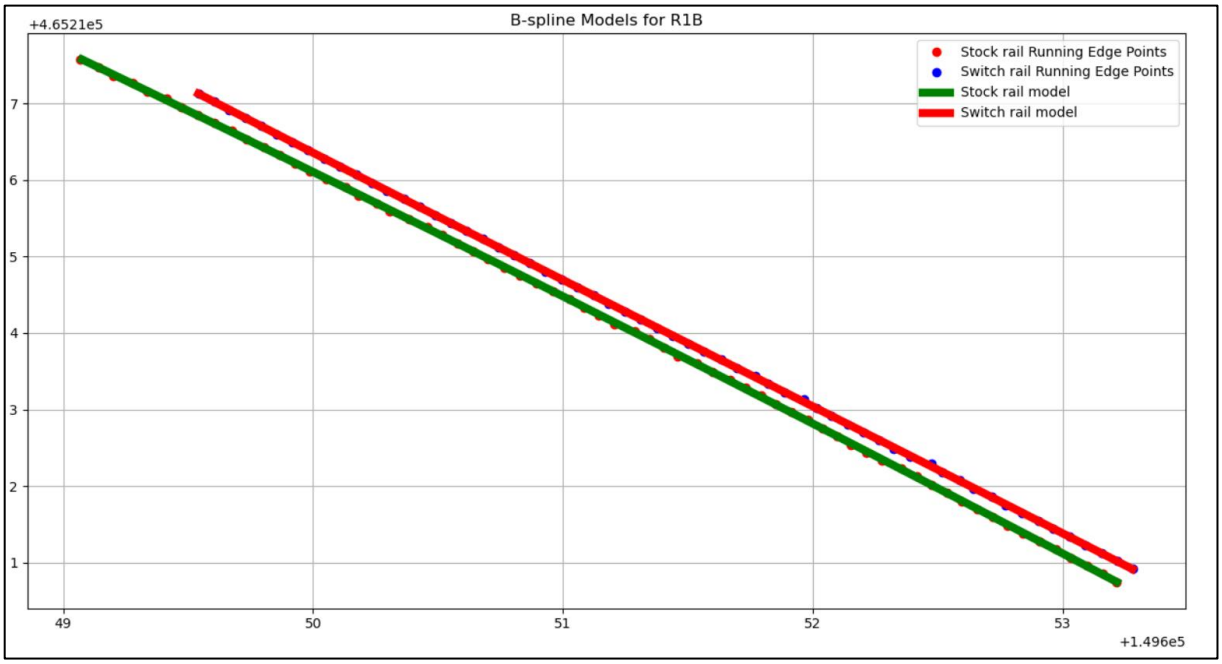


Figure 5-7: Selected running edge points shown on both stock and switch rails. The stock rail is shown in green and the switch rail in blue. The running edge points are shown in red.

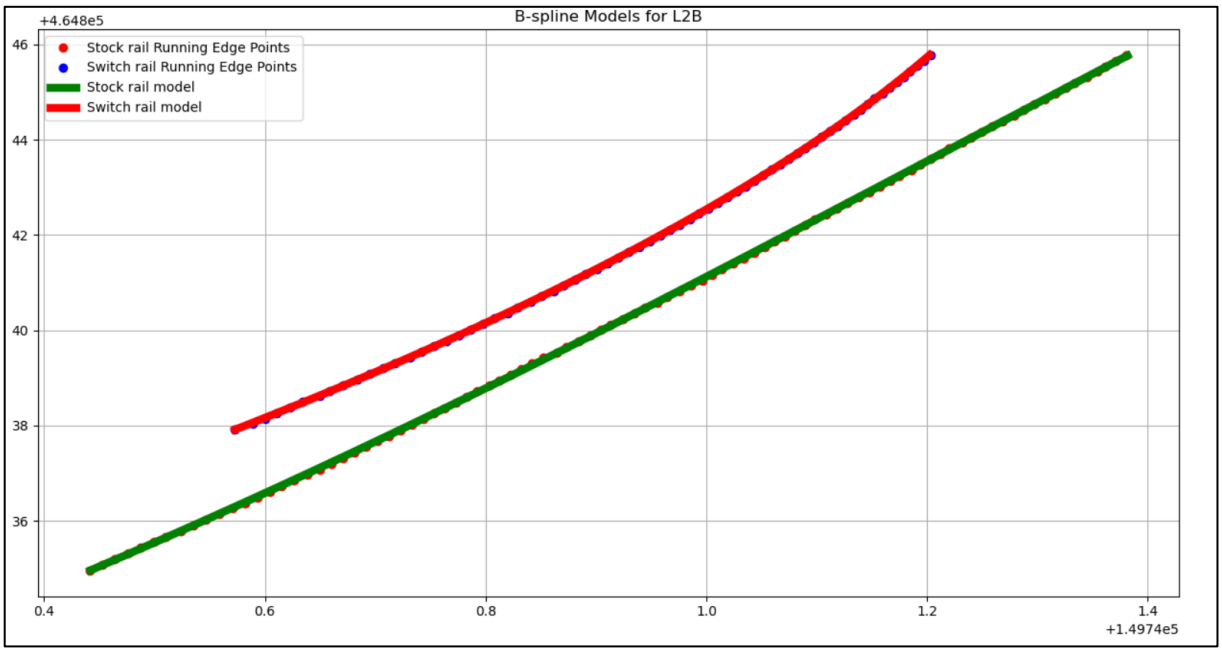
5.3. Measuring the flangeway

A B-spline curve is fitted to rail running edge points to generate a continuous model for each stock rail and switch rail. Each curve is evaluated to have continuous dense sets of points depending on the length of running edge points. The selected running edge points for all the switch rails in the region of interest covered a length of about 7 m. Therefore, the splines models generated for switch rails had 70000 points, with a point-to-point interval of about 0.1 mm. The number of points on the stock rail models differs per strip. Examples of the models generated are shown in Figure 5-8 below. Figure 5-8 (a) shows an example of models (for R1B) where the rails are fairly straight, while Figure 5-8 (b) depicts the model for a slightly curved switch rail. It is observed that the B-spline model fitted well to the shape of the rails. Furthermore, on the models in both images, the running edge points are plotted on the models to show how the B-spline mitigated the limitations relating to the incorrect points. It can be seen that the smooth models are generated without wiggling connections to the wrong running edge points. Therefore, these incorrect

points are seen to be slightly off the curve models generated. This is expected since a B-spline is not designed to connect the points exactly but to produce a mathematically smooth curve. However, points that are accurately determined overlap nicely with the spline model.



(a)



(b)

Figure 5-8: B-Spline models generated for stock rail and its switch rail for two different strips. (a) Models for a strip where rails are fairly straight. (b) Models for a strip where the switch rail is slightly curved.

Distances are computed between the stock rail and its switch rail using the spline models generated. Figure 5-9 indicates the distances computed for two of the strips from different epochs. The distances are computed on every location of the rail models in the area of interest, which has a length of about 7 m. Since the model used had continuous points with an interval of about 0.1 mm, it implies that distances can be computed in a continuous manner between the stock rail and its switch rail at this interval. However, for visualisation purposes, these computed distances are shown at an interval of 0.2 m on the graph below. Also, it should be noted that points can be selected arbitrarily at any location on the models to determine the distance between the two rails at that location. The aim of showing the distances for the two epochs is not to compare the distances but to give an indication of how the distances can be computed in a continuous manner. For this reason, only one of the results in this section is presented. From the distances computed per strip, the minimum distance is determined. This minimum distance's position is similarly determined for all the strips used; therefore, only one figure is shown in Figure 5-10.

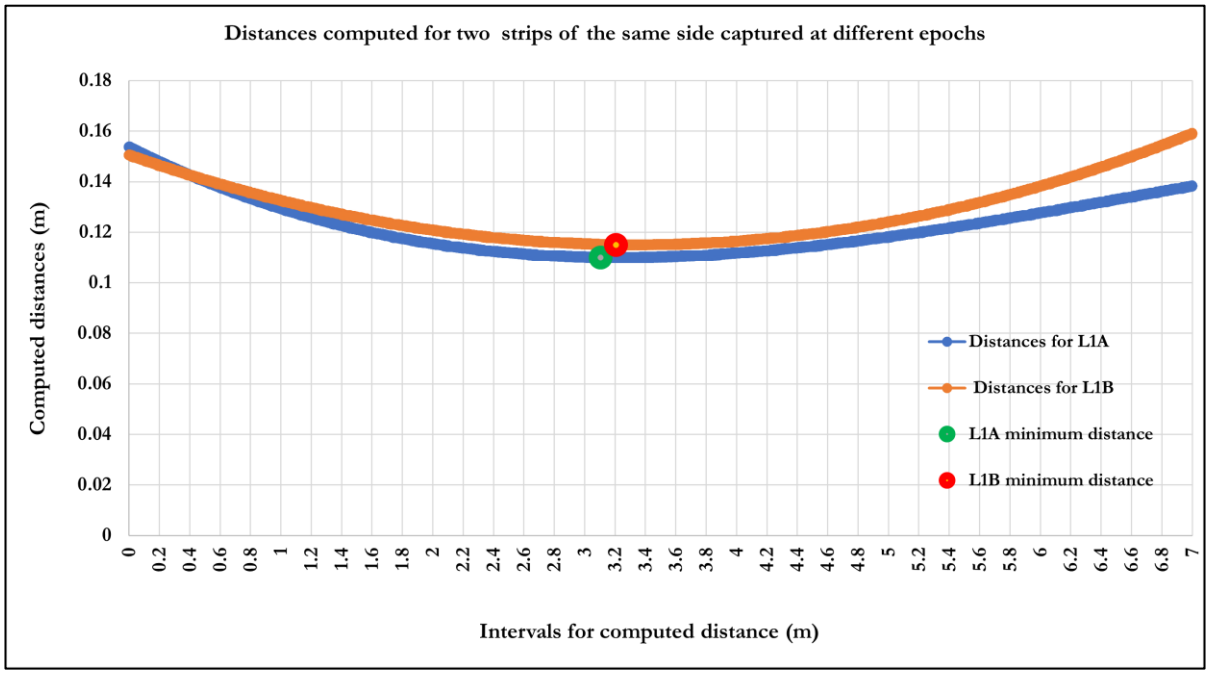


Figure 5-9: Results showing the distances computed in a continuous manner between the stock rail and its switch rail for two strips.

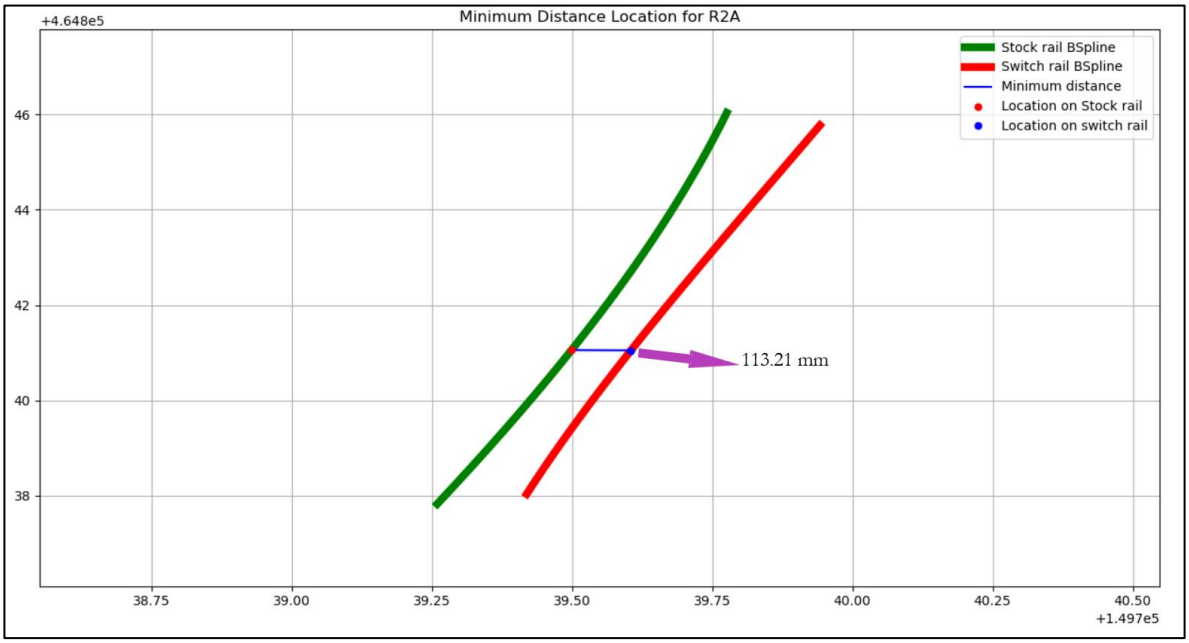


Figure 5-10: Image showing the location where the minimum distance was obtained

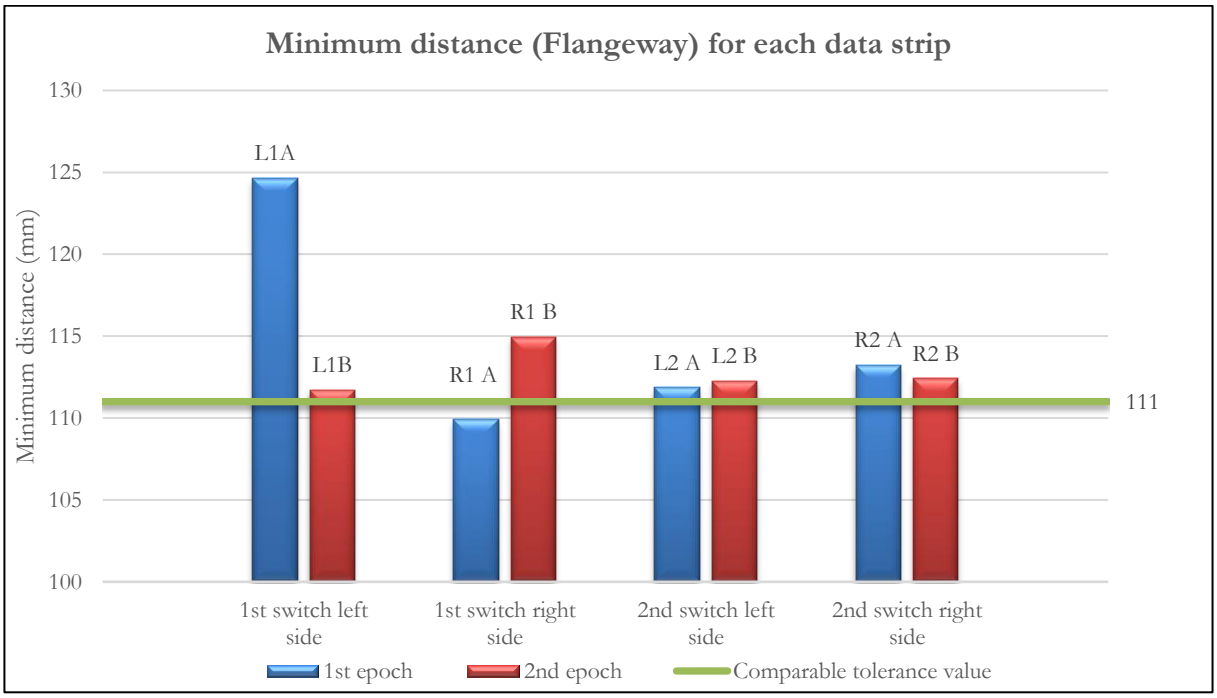


Figure 5-11: Minimum distances (Flangeway) obtained per each strip. The solid green line shows the tolerance value that is comparable with the data. The comparable tolerance value (111mm) is the actual tolerance value (41mm) plus the approximate width of the switch rail profile (70mm).

Figure 5-11 above shows the minimum distance computed for each strip. The distances are plotted for each strip based on the switch and the epoch it was captured. These minimum distances are not the actual flangeway but the adjustment measurement made at the inner running edge to the outer running edge of the stock and switch rails, respectively. Therefore, an adjustment is made to the tolerable flangeway value (41mm) so that the computed flangeways can be compared to this value. Fugro provided the tolerable flangeway value based on Prorail's technical specifications for switches (Check Appendix 2).

The comparable tolerable value is obtained by adding the tolerable flangeway value to the width of the switch rail. However, since we do not know the actual width of the switch rail, an approximation based on the width of the stock rail (70mm) is used. The comparable tolerance value obtained is 111 mm, and this is plotted as a solid green line on the graph. The deviations of computed flangeway per strip to the comparable tolerable value are shown in Table 5-2. From the table, it is observed, some of the deviations are small (in millimetres), while one strip has a relatively high deviation value (approximately 1.36 cm). These deviations are only to demonstrate how the computed flangeway can be used to make further assessments on the rails but not to make conclusions on the state of the rails.

Table 5-2: Table showing how the computed flangeway deviates from the comparable tolerance value

Switch	Data strip	Deviation from tolerable value (mm)
1st switch left side	L1A	13.63
	L1B	0.72
1st switch right side	R1A	-1.07
	R1B	3.95
2nd switch left side	L2A	0.86
	L2B	1.27
2nd switch right side	R2A	2.21
	R2B	1.40

6. DISCUSSION

6.1. Extraction of stock rail and switch rail

With the SOR filter, the standard deviation used by the algorithm is defined based on the nature of the outlier as well as their density. For strips with less sparse outliers, the defined standard deviation removes these outliers effectively, even in areas with gaps without removing the rail head points. However, for strips with sparse and dense clusters of outliers, there is a trade-off between preserving the points around the rail head and removing the outliers. This is because the strips had data gaps; hence increasing the filter's aggressivity to remove the dense points results in the removal of points in the sections on the railhead (see Appendix 3). For such data strips, some dense and sparse non-isolated outliers remained. This is because these outliers have a similar distribution to points on the rail head, and with the defined standard deviation, the algorithm could not differentiate them as outliers. However, the dense outliers are easily removed after the connected component segmentation. As the remaining non-isolated sparse clusters of outliers are few in number, they do not have any influence on the further processes used in this study. It may be said that the algorithm is more efficient in removing sparse outliers, which are very isolated than other forms of outliers. Future studies can explore the local projection method proposed by Ning et al. (2018) to remove the non-isolated outliers if they are to be removed completely.

While implementing the grid-based elevation filtering method for removing the ground points, an oriental bounding box is used to generate the grid. Since the data strips used are narrow, the oriental bounding box used fitted well to the shape of the strip preventing the creation of grid cells with no points. With all the grid cells containing points, no further analysis was required to select grids cells with only points. Also, because the strips are narrow, rectangular grid cells are used instead of square grid cells, as used in the study of Oude Elberink et al. (2013) and Arastounia (2012). Using a rectangular grid resulted in few cells with no rail points. Points in these cells did not satisfy the filtering condition and are not removed. These points are separated from the rail region; therefore, further analysis is not performed to eliminate them. This is because the points formed separate segments, and the connected component could easily segment them out to achieve a good segmentation of the rails. Defining the grid cell's width based on the half-width of the bounding box resulted in a big cell size for some of the strips. This is because the width differs per strip. This resulted in having multiple rails in some of the cells. For strips where this occurred, some points on the web and the tip of the switch rail remained. These remaining points could not be removed by segmentation because they were connected to the rail head. However, they were eliminated by a buffer during the determination of the running edge point. (check section 4.3.2). Future studies should explore better ways of defining the grid size to prevent these limitations.

By using connected components, a grid step size of 0.01 m is chosen based on experimentation and yielded good segmentation results. The algorithm employed does not allow to set grid step size; this is determined at an octree level depending on the size of the point clouds. Using the octree is a recommended way because of the algorithm speed. However, it implies that the octree level needs to be chosen per each strip to maintain the above-mentioned grid step size. Considering the number of strips used in this study, performing this manually is a tedious task and can even be impractical if a larger number of strips are to be used. Future studies can automate the whole approach by using the conditional Euclidean distance clustering algorithm in the Point Cloud Library (PCL). However, in the case where technical issues do not permit PCL use, the strips should be trimmed to the same extent so that once the right octree level is achieved, it can be applied to the rest of the data.

6.2. Determining the running edge of the rails

This study utilised prior knowledge of the rail head profile to determine the running edge positions along the rail. During the implementation of this stage, this study took advantage of the orientation angles in the trajectory data obtained during the MMS survey to determine the directions of the rail. With most of the strips having data gaps and missing points around the rail head, a buffer is designed to select a number of scanlines and project the points on these scanlines to 2D to increase the point density for accurate determination of the running edge. However, this method did not work very well when the sections with missing point clouds are longer on the rail. This implies there will be no good scanlines selected in the buffer to supplement the others to achieve accurate running edge points for such sections. As stated earlier in section 5.2, the use of the stock rail profile for the switch rail resulted in wrong running edge points at the tip of the switch rail. Therefore, in future work, the correct profile for the switch rail should be used. This is expected to increase the accuracy of the individual distances measured between the stock rail and the corresponding switch rail, but it is unlikely to make a big impact on the flangeway measurement as the flangeway resides away from the thin tip of the switch rail.

6.3. Measuring the flangeway

In this study, the advantages of the B-spline curve, which includes its ability to create continuity between data with gaps as well as deal with outliers, are utilised. Although some of the running edge points are determined incorrectly, using the B-spline mitigated the effect of these points and created smooth curve models for both rails. Another advantage of using the B-spline is that it generated models with dense points having a discretization of 0.1 mm. This allowed for the computation of distances at every location between rails, although the curvature of these two rails is different. By computing the distances in a continuous manner, the flangeway for each strip is determined. The computed minimum distances per strip are not the actual flangeway. Also, the comparable tolerable value used for finding the deviation of the flangeway is not based on the actual width of the switch rail. Therefore, the deviations may be different from the actual deviation in the flangeway. It should be noted that the study does not aim to make conclusions on if there are deformations in the flangeway based on the deviations. To do this, one will require the maximum tolerable value and the actual width of the switch rail profile. Therefore, future studies can explore this further.

7. CONCLUSION AND RECOMMENDATIONS

7.1. Conclusion

This study proposes a method to measure the flangeway of railway switches from point clouds. The main objective of the research is to develop an algorithm that turns the discrete point clouds into continuous models, which are then used to determine the flangeway. The objective is achieved in a three-stage methodology.

The proposed method begins with extracting the rails as separate subsets of points from the point cloud strips. As segmentation is employed to extract the rail points, the first stage of the method removes outliers present in the data and ground points that created connections between the rails. The removal of these points is essentially helpful to achieve a good segmentation of the rails. The segmentation of the rails is achieved using a connected components algorithm. By using this approach, points on the railhead for each stock rail and switch rail are well segmented into separate subsets of points. However, in few cases, some of the points on the rail web and tip of the switch rail are also segmented as rail head points. This because these points are also connected to the rail head.

Following the segmentation, the running edge positions along the rails are determined. This is achieved by using prior knowledge about the designed stock rail profile to define a template which is then matched to 2D projected points of selected scanlines. The matching is done using a brute force shape matching approach. The results obtained indicate that most of the running edge positions are accurately determined for the rails except for areas where there were many missing rail head points.

The determined running edge points are used as the base data to generate models for each stock rail and its corresponding switch rail. B-spline is employed to fit a curve to the running edge points, which results in models with dense sets of points for each stock rail and switch rail. Although there are some running edge points that deviate from the actual position, the use of the B-spline curve curbed the influence of these points by generating smooth models while ignoring these erroneous points. The resulting models for the stock rail and its switch rail are used to compute distances in a continuous manner between the two rails. The flangeway, which is the minimum distance between these two rails, is then determined from the computed distances. The minimum distance determined is not the actual flangeway but a proxy distance based on the inner running edge of the stock rail and the outer running edge of the switch rail. However, the actual flangeway can easily be determined by subtracting a constant distance value that is already known from the switch rail profile design from the computed flangeway.

The proposed algorithm in this study has demonstrated its capability to determine the flangeway of switches from points clouds and can therefore be used for commercial purposes.

7.2. Recommendations

- Future studies should improve the proposed algorithm by using the right switch rail profile when determining the running edge points for the switch rail. By doing this, another inspection parameter which the maximum distance in the opening and occurs at the point machine location on the switch (check Figure 1-2), can be determined. Further improvement can also be made to determine vertical inspection parameters in this region.

- The proposed methodology is applied to a simple switch whose rails are almost straight. Future studies should explore the methods described in this study to other types of switches as well as very curvy rails.

- Future studies can further explore this method using the known information on the switch rail profile design to obtain the actual flangeway and then assess the accuracy using field reference data.

- To use the method for commercial purposes, railway infrastructure companies should utilise the known information on the switch rail profile design to obtain the actual flangeway. Future studies can also use this information and the maximum tolerable value to determine if there are deformations in the rails.

7.3. Research questions answered

1. What method can be used to extract stock rail and switch rail as separate rails?

   Connected components segmentation preceded by removing outliers and ground points is used to extract the stock and switch rails into separate subsets of points.

2. How will the running edge positions of the rails be determined?

   With the projection of selected scanlines to 2D and an a priori known template of the rail head, a brute force approach is used to determine the positions of the running edge on the rails.

3. Can the distances be determined in a continuous manner between the two rails?

   Yes. By using a B-spline curve, spatially continuous models are generated for a stock rail and its counterpart, the switch rail. These models are used to compute distances at every location for the rails, as shown in Figure 5-9. This helps to determine the flangeway and the position on the rails where the flangeway is determined.

4. Does the method work well to determine the flangeway?

   The proposed method is capable of determining the minimum distance between the inner running edge of the stock rail and the outer running edge of the switch rail. Once this minimum distance is determined, then knowledge of the switch rail profile can be utilised to compute the actual flangeway for the switches. Therefore, it can be said that the method works well to determine the flangeway.

   .

LIST OF REFERENCES

Arastounia, M. (2012). *Automatic Classification of LiDAR Point Clouds in A Railway Environment (Masters' thesis)*. (Masters' Thesis). Retrieved from https://webapps.itc.utwente.nl/librarywww/papers_2012/msc/gfm/arastounia.pdf

Arastounia, M. (2016). Automated As-Built Model Generation of Subway Tunnels from Mobile LiDAR Data. *Sensors*, *16*(9), 1486–1506. https://doi.org/10.3390/s16091486

Belkhouche, Y., Duraisamy, P., & Buckles, B. (2015). Graph-connected components for filtering urban LiDAR data. *Journal of Applied Remote Sensing*, *9*(1), 096075–096088. https://doi.org/10.1117/1.JRS.9.096075

Bemment, S. D., Ebinger, E., Goodall, R. M., Ward, C. P., & Dixon, R. (2017). Rethinking rail track switches for fault tolerance and enhanced performance. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, *231*(9), 1048–1065. https://doi.org/10.1177/0954409716645630

Biosca, J. M., & Lerma, J. L. (2008). Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods. *ISPRS Journal of Photogrammetry and Remote Sensing*, *63*(1), 84–98. https://doi.org/10.1016/j.isprsjprs.2007.07.010

Bureick, J., Alkhatib, H., & Neumann, I. (2016). Robust Spatial Approximation of Laser Scanner Point Clouds by Means of Free-form Curve Approaches in Deformation Analysis. *Journal of Applied Geodesy*, *10*, 27–35. https://doi.org/10.1515/jag-2015-0020

Bureick, J., Neuner, H., Harmening, C., & Neumann, I. (2016). Curve and surface approximation of 3D point clouds. *AVN Allgemeine Vermessungs-Nachrichten*, *123*(11–12), 315–327. Retrieved from www.researchgate.net/publication/311794543_Curve_and_surface_approximation_of_3D_point_clouds

Che, E., Jung, J., & Olsen, M. J. (2019). Object recognition, segmentation, and classification of mobile laser scanning point clouds: A state of the art review. *Sensors (Switzerland)*, *19*(4), 810–852. https://doi.org/10.3390/s19040810

Che, E., & Olsen, M. J. (2018). Multi-scan segmentation of terrestrial laser scanning data based on normal variation analysis. *ISPRS Journal of Photogrammetry and Remote Sensing*, *143*, 233–248. https://doi.org/10.1016/j.isprsjprs.2018.01.019

Chen, B., Shi, S., Sun, J., Gong, W., Yang, J., Du, L., … Chen, B. (2019). Hyperspectral lidar point cloud segmentation based on geometric and spectral information. *Optics Express*, *27*(17), 24043–24059. https://doi.org/10.1364/oe.27.024043

Díaz Benito, D. (2012). *Automatic 3D modelling of train rails in a lidar point cloud*. Retrieved from http://www.itc.nl/library/papers_2012/msc/gfm/diazbenito.pdf

Elberink, S., & Khoshelham, K. (2015). Automatic Extraction of Railroad Centerlines from Mobile Laser Scanning Data. *Remote Sensing*, *7*(5), 5565–5583. https://doi.org/10.3390/rs70505565

Farin, G. (2002). B-Spline Curves. In B. A. Barsky (Ed.), *Curves and Surfaces for CAGD* (pp. 119–146). https://doi.org/10.1016/B978-155860737-8/50008-9

Fugro. (2021). RILA 3.0. Retrieved from https://www.fugro.com/about-fugro/our-expertise/innovations/rila

Hamadache, M., Dutta, S., Olaby, O., Ambur, R., Stewart, E., & Dixon, R. (2019). On the Fault Detection and Diagnosis of Railway Switch and Crossing Systems: An Overview. *Applied Sciences*, *9*(23), 5129–5160. https://doi.org/10.3390/app9235129

Harmening, C. (2020). *Spatio-temporal deformation analysis using enhanced B-spline models of laser scanning point clouds (Doctoral dissertation)*. Retrieved from https://doi.org/10.34726/hss.2020.57320

Hesami, R., BabHadiashar, A., & HosseinNezhad, R. (2010). Range segmentation of large building exteriors: A hierarchical robust approach. *Computer Vision and Image Understanding*, *114*(4), 475–490.

https://doi.org/10.1016/j.cviu.2009.12.004

International Union of Railways, [UIC]. (2015). *Operational failure modes of Switches and Crossings*. Retrieved from http://www.capacity4rail.eu/IMG/pdf/c4r_-_d131_-_operational_failure_modes_of_scs_public_.pdf

Jwa, Y., & Sonh, G. (2015). KALMAN FILTER BASED RAILWAY TRACKING FROM MOBILE LIDAR DATA. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *II-3/W5*(3W5), 159–164. https://doi.org/10.5194/isprsannals-II-3-W5-159-2015

Kaewunruen, S. (2014). Monitoring structural deterioration of railway turnout systems via dynamic wheel/rail interaction. *Case Studies in Nondestructive Testing and Evaluation*, *1*, 19–24. https://doi.org/10.1016/j.csndt.2014.03.004

Kaewunruen, S., & Lian, Q. (2019). Digital twin aided sustainability-based lifecycle management for railway turnout systems. *Journal of Cleaner Production*, *228*, 1537–1551. https://doi.org/10.1016/j.jclepro.2019.04.156

Karunathilake, A., Honma, R., & Niina, Y. (2020). Self-Organized Model Fitting Method for Railway Structures Monitoring Using LiDAR Point Cloud. *Remote Sensing*, *12*(22), 3702–3717. https://doi.org/10.3390/rs12223702

Lindenbergh, R., & Pietrzyk, P. (2015). Change detection and deformation analysis using static and mobile laser scanning. *Applied Geomatics*, *7*(2), 65–74. https://doi.org/10.1007/s12518-014-0151-y

Lou, Y., Zhang, T., Tang, J., Song, W., Zhang, Y., & Chen, L. (2018). A Fast Algorithm for Rail Extraction Using Mobile Laser Scanning Data. *Remote Sensing*, *10*(12), 1998–2019. https://doi.org/10.3390/rs10121998

Lu, X., Yao, J., Tu, J., Li, K., Li, L., & Liu, Y. (2016). PAIRWISE LINKAGE FOR POINT CLOUD SEGMENTATION. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *3*, 201–208. https://doi.org/10.5194/isprs-annals-III-3-201-2016

Mikrut, S., Kohut, P., Pyka, K., Tokarczyk, R., Barszcz, T., & Uhl, T. (2016). Mobile Laser Scanning Systems for Measuring the Clearance Gauge of Railways: State of Play, Testing and Outlook. *Sensors*, *16*(5), 683–707. https://doi.org/10.3390/s16050683

Mukupa, W., Roberts, G. W., Hancock, C. M., & Al-Manasir, K. (2016). A review of the use of terrestrial laser scanning application for change detection and deformation monitoring of structures. *Survey Review*, *49*(353), 99–116. https://doi.org/10.1080/00396265.2015.1133039

Neuner, H., Holst, C., & Kuhlmann, H. (2016). Overview on current modelling strategies of point clouds for deformation analysis. *AVN Allgemeine Vermessungs-Nachrichten*, *123*(11–12), 328–339. Retrieved from https://www.researchgate.net/publication/311794822_Overview_on_current_modelling_strategies_of_point_clouds_for_deformation_analysis

Niina, Y., Honma, R., Honma, Y., Kondo, K., Tsuji, K., Hiramatsu, T., & Oketani, E. (2018). AUTOMATIC RAIL EXTRACTION AND CELARANCE CHECK WITH A POINT CLOUD CAPTURED BY MLS IN A RAILWAY. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLII–2*(2), 767–771. https://doi.org/10.5194/isprs-archives-XLII-2-767-2018

Ning, X., Li, F., Tian, G., & Wang, Y. (2018). An efficient outlier removal method for scattered point cloud data. *PLOS ONE*, *13*(8), 1–22. https://doi.org/10.1371/journal.pone.0201280

Oude Elberink, S., Khoshelham, K., Arastounia, M., & Díaz Benito, D. (2013). Rail Track Detection and Modelling in Mobile Laser Scanner Data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *2*(5W2), 223–228. https://doi.org/10.5194/isprsannals-II-5-W2-223-2013

Pastucha, E. (2016). Catenary System Detection, Localization and Classification Using Mobile Scanning Data. *Remote Sensing*, *8*(10), 801–822. https://doi.org/10.3390/rs8100801

Rabbani, T., van den Heuvel, F. a, & Vosselman, G. (2006). Segmentation of point clouds using smoothness constraint. *International Archives of Photogrammetry, Remote Sensing and Spatial Information*

Sciences - Commission V Symposium 'Image Engineering and Vision Metrology', *36*(5), 248–253. Retrieved from http://www.isprs.org/proceedings/XXXVI/part5/paper/RABB_639.pdf

Rusu, M. F. (2015). *AUTOMATION OF RAILWAY SWITCH AND CROSSING INSPECTION (Doctoral dissertation)*. Retrieved from https://studylib.net/doc/6774949/railway-automation---automatic-switch-and-crossing-inspection

Soni, A., Robson, S., & Gleeson, B. (2014). Extracting rail track geometry from static terrestrial laser scans for monitoring purposes. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *40*(5), 553–557. https://doi.org/10.5194/isprsarchives-XL-5-553-2014

Standards, B. (2003). *Railway applications Track Switches and crossings (13232-1:2003)*. Retrieved from http://106.38.59.21:8080/userfiles/843240d8c0034bbb9d313aff660b4f0e/files/teckSolution/2019/07/BS EN 13232-1-2003.pdf

Trevor, A. J. B., Gedikli, S., Rusu, R. B., & Christensen, H. I. (2013). Efficient organized point cloud segmentation with connected components. *Semantic Perception Mapping and Exploration (SPME)*, 1–6. Retrieved from https://cs.gmu.edu/~kosecka/ICRA2013/spme13_trevor.pdf

Turabimana, P., & Nkundineza, C. (2020). Development of an On-Board Measurement System for Railway Vehicle Wheel Flange Wear. *Sensors*, *20*(1), 303–323. https://doi.org/10.3390/s20010303

Walton, G., Delaloye, D., & Diederichs, M. S. (2014). Development of an elliptical fitting algorithm to improve change detection capabilities with applications for deformation monitoring in circular tunnels and shafts. *Tunnelling and Underground Space Technology*, *43*, 336–349. https://doi.org/10.1016/j.tust.2014.05.014

Wang, Q., Tan, Y., & Mei, Z. (2019). Computational Methods of Acquisition and Processing of 3D Point Cloud Data for Construction Applications. *Archives of Computational Methods in Engineering*, *27*(2), 479–499. https://doi.org/10.1007/s11831-019-09320-4

Xu, X., Yang, H., & Kargoll, B. (2019). Robust and automatic modeling of tunnel structures based on terrestrial laser scanning measurement. *International Journal of Distributed Sensor Networks*, *15*(11), 1–9. https://doi.org/10.1177/1550147719884886

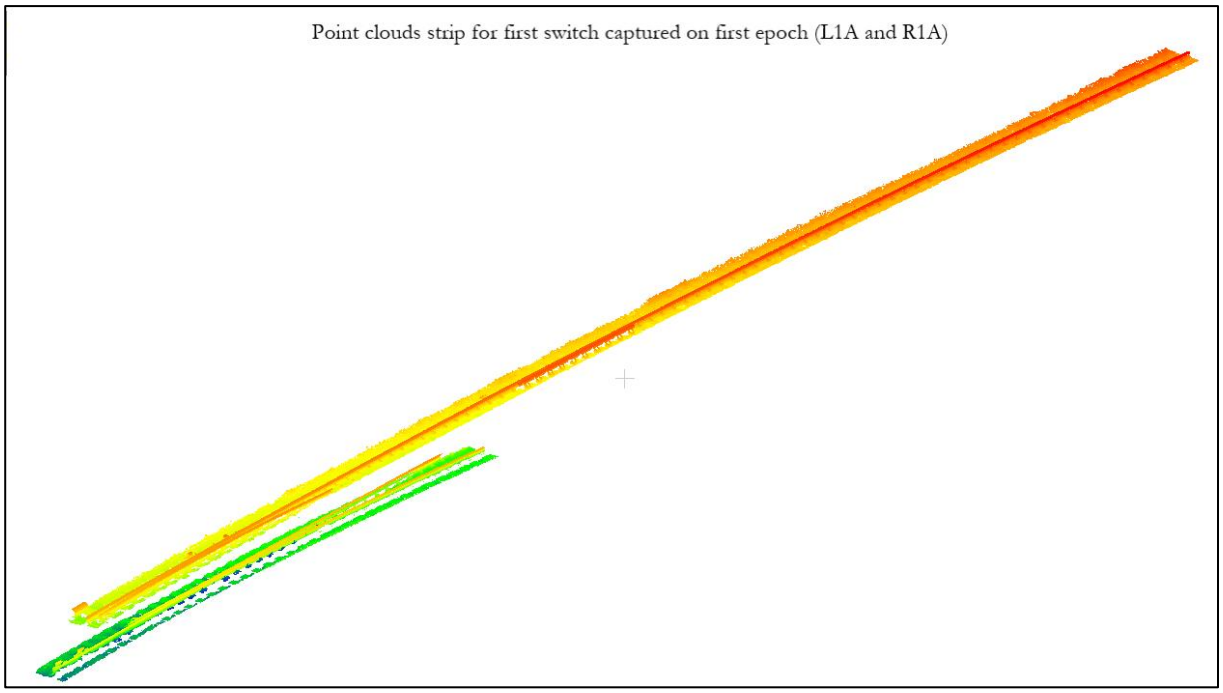
Yang, B., & Fang, L. (2014). Automated Extraction of 3-D Railway Tracks from Mobile Laser Scanning Point Clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*(12), 4750–4761. https://doi.org/10.1109/JSTARS.2014.2312378

Zhao, X., Kargoll, B., Omidalizarandi, M., Xu, X., & Alkhatib, H. (2018). Model Selection for Parametric Surfaces Approximating 3D Point Clouds for Deformation Analysis. *Remote Sensing*, *10*(4), 634–655. https://doi.org/10.3390/rs10040634

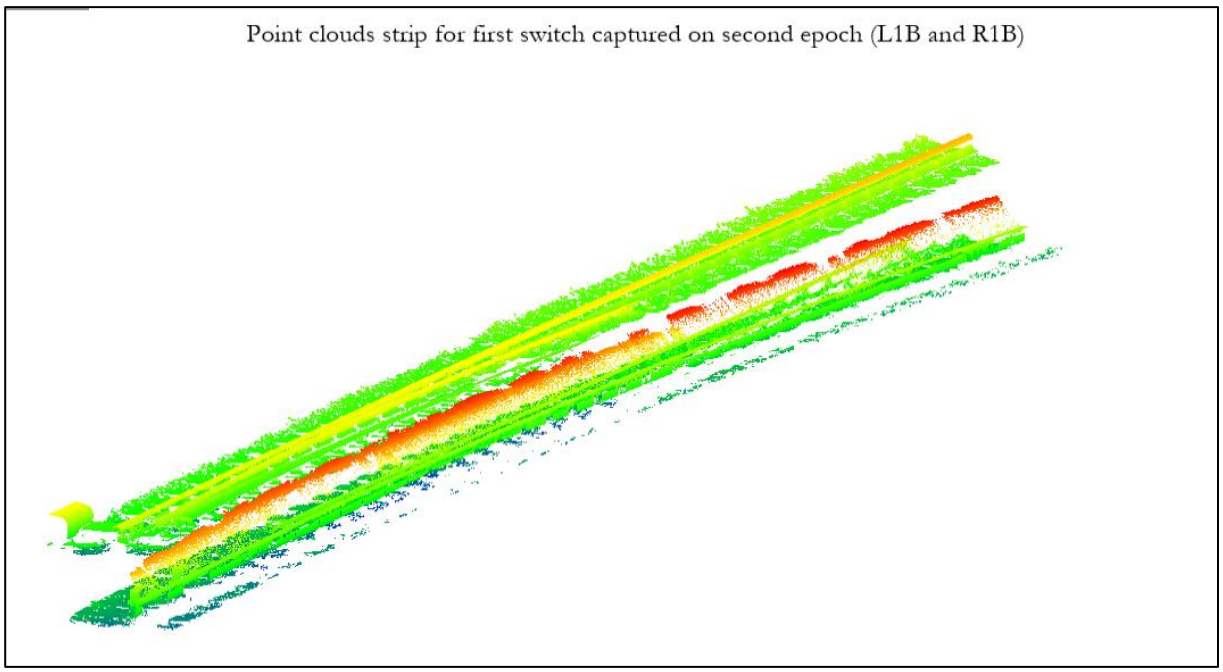
Zou, R., Fan, X., Qian, C., Ye, W., Zhao, P., Tang, J., & Liu, H. (2019). An Efficient and Accurate Method for Different Configurations Railway Extraction Based on Mobile Laser Scanning. *Remote Sensing*, *11*(24), 2929–2950. https://doi.org/10.3390/rs11242929

APPENDIX 1

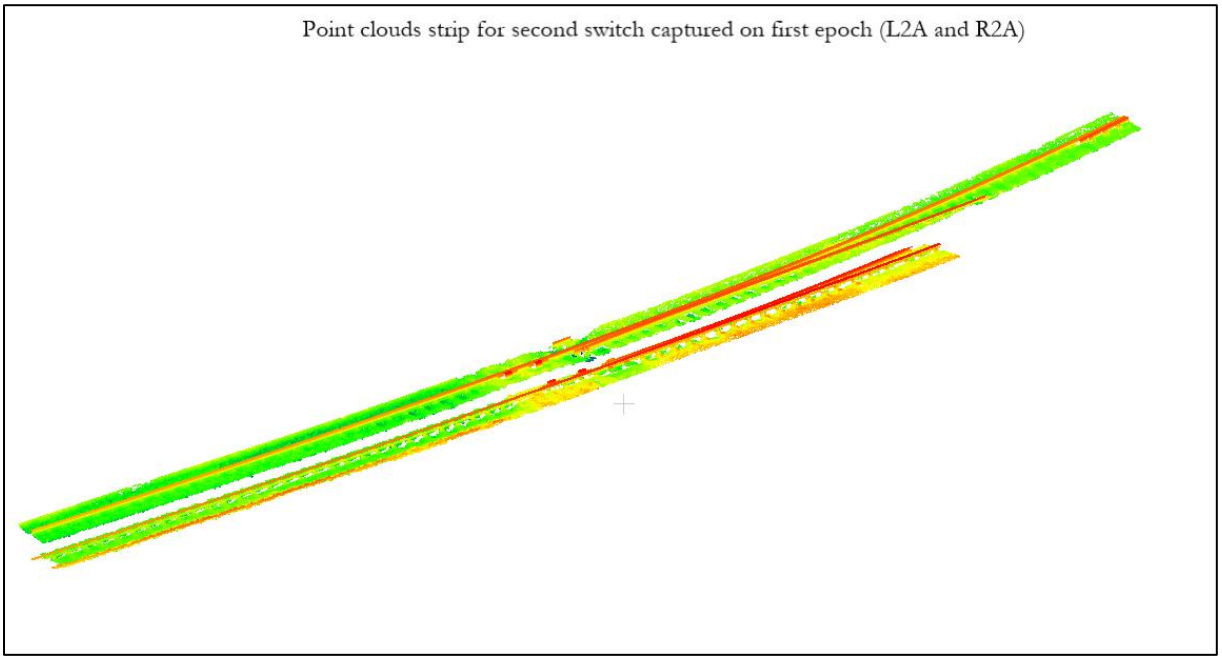
Data Strips Used



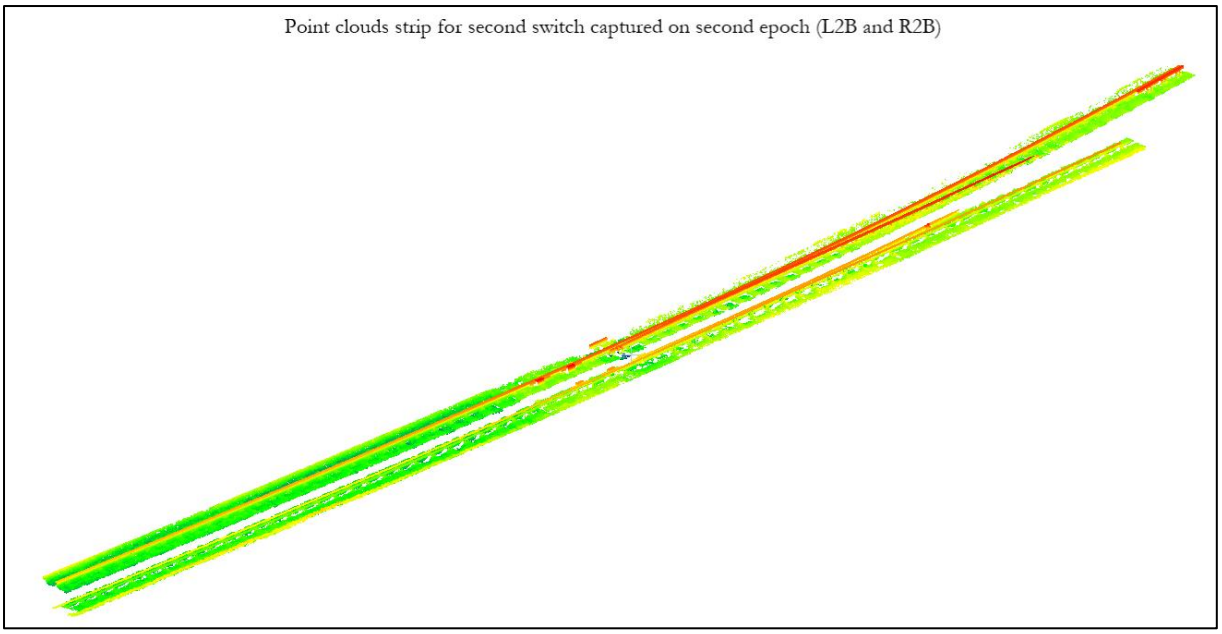
(a)



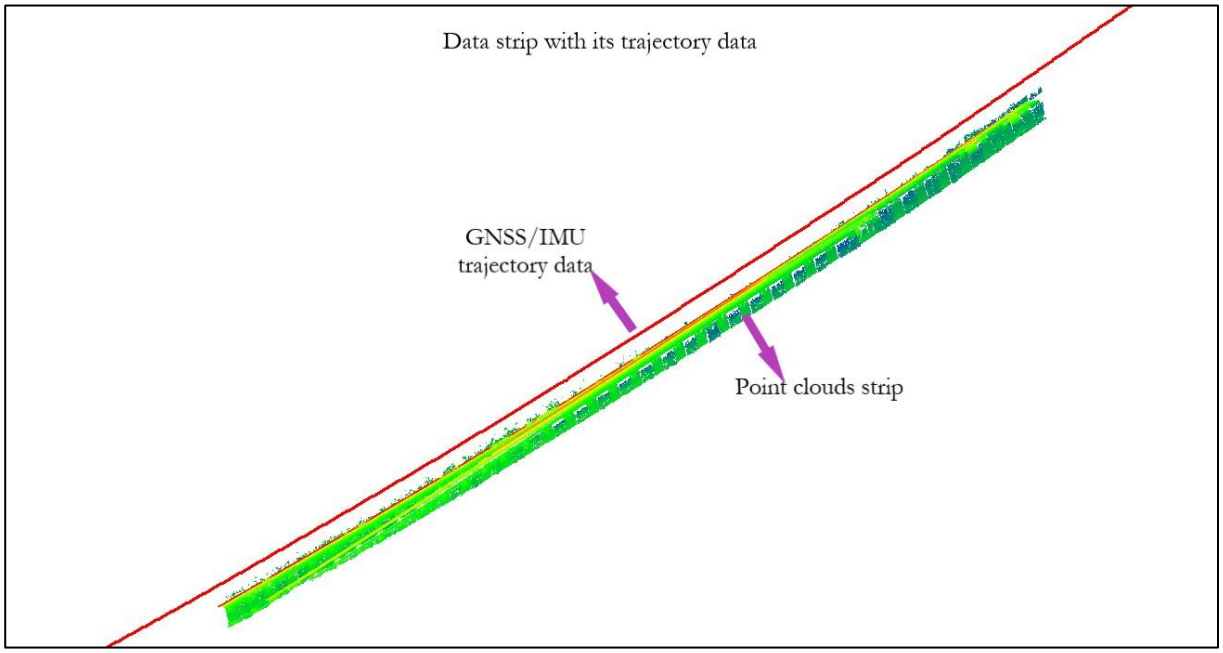
(b)


Point clouds strip for second switch captured on first epoch (L2A and R2A)

(c)


Point clouds strip for second switch captured on second epoch (L2B and R2B)

(d)


Data strip with its trajectory data
GNSS/IMU
trajectory data
Point clouds strip

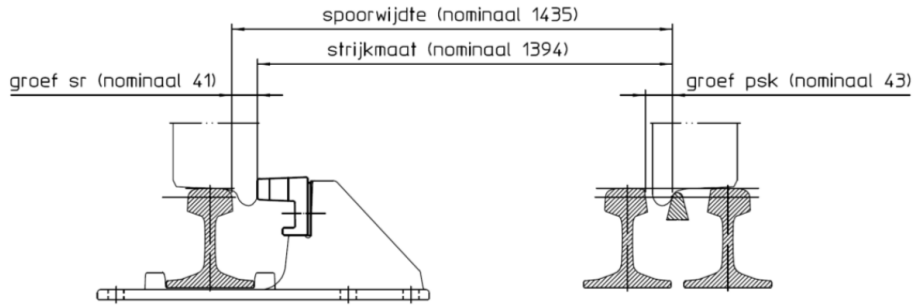
(e)

APPENDIX 2

Tolerable Minimum Flangeway Value

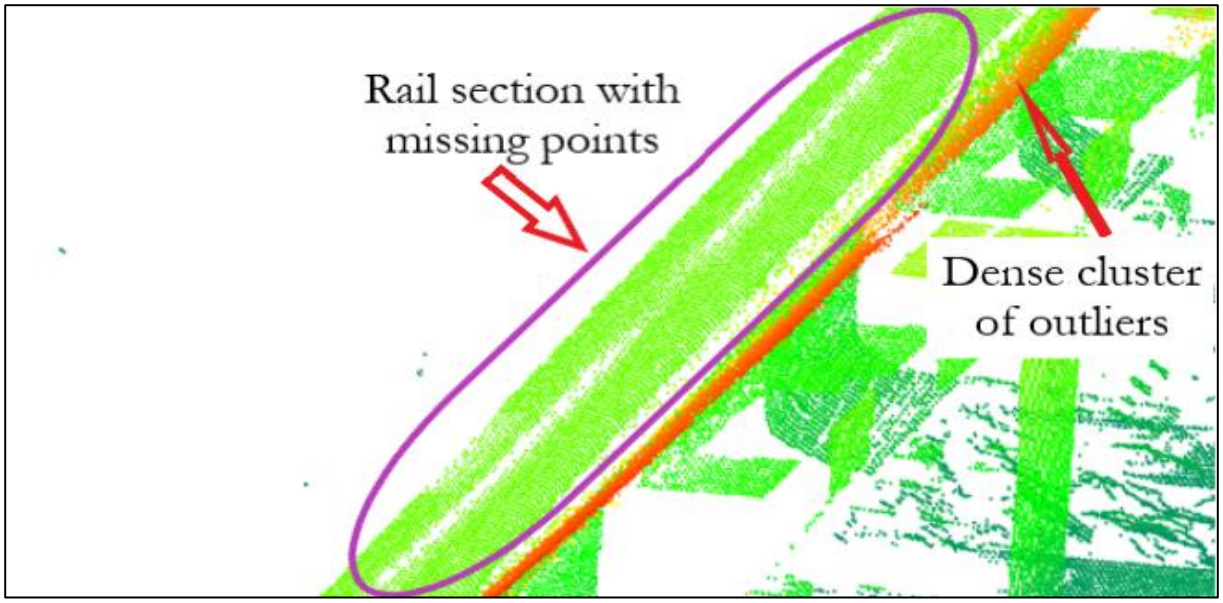
ProRail OHD00022-1

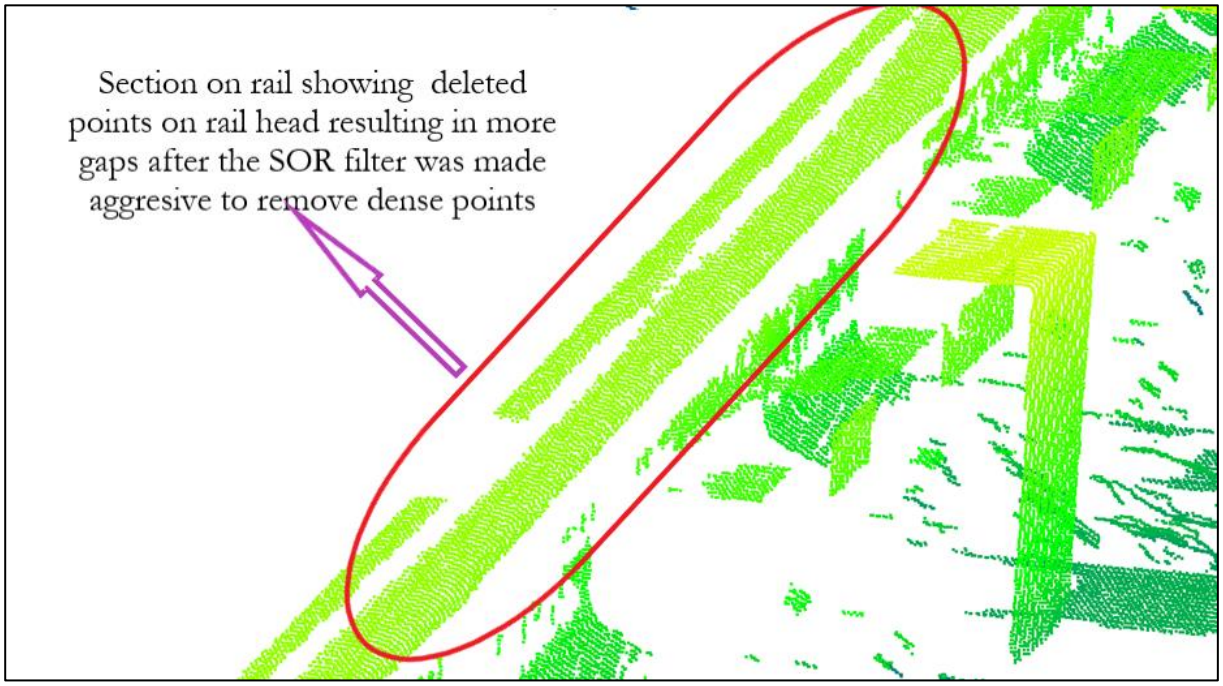
Instandhoudingsspecificatie wissels en kruisingen (geometrie)



Nominaal is de strijkmaat 1394 mm. Uitgangspunt hierbij is een nominale spoorwijdte van 1435 mm en een groefbreedte van 41 mm tussen strijkregel en naastgelegen spoorstaaf. De nominale groefbreedte in een punt- of kruisstuk is meestal 43 mm. Uitzondering is het puntstuk 54 E1 1:4,5 dat sinds 2004 een groef van 41 mm breed heeft.

APPENDIX 3


Rail section with
missing points
Dense cluster
of outliers


Section on rail showing deleted
points on rail head resulting in more
gaps after the SOR filter was made
aggresive to remove dense points