ACCESSIBILITY: Open

OsloMet – Oslo Metropolitan University

Department of Civil Engineering & Energy Technology

Section of Civil Engineering

MASTER THESIS

TITLE OF REPORT	DATE
Automatic BIM-model creation of bridge structure using 3D-laser scanning, algorithms and parametric modeling	25.05.2022
	PAGES / ATTACHMENTS 78/3
AUTHOR(S) Faryad Daven & Shahin Ezati	SUPERVISOR(S) Daguang Han & Gro Markeset

IN COLLABORATION WITH	CONTACT PERSON

SUMMARY / SYNOPSIS

Bridges have a crucial and huge role in the safety of the public. An important aspect in maintaining the safety of a bridge structure is to have access to a BIM-model that can be used in analysis and facility management. Creation of such a model for long-span spatial curved structures with circular pipe components quickly and accurately depends on measuring the overall alignment of such structures, which is a problem in building industry today. This thesis has developed a method to measure overall alignment of long-span spatial curved structures with circular pipe components to generate a BIM-model automatically. 3D-laser scanning has been implemented to collect data from a real case study which is a pedestrian bridge located in Oslo, Norway. The data has been imported to MATAB to extract geometry information using algorithms. The information extracted are imported into Dynamo to create a parametric driven BIM-model. Finally, the deviation between the BIM-model and the point cloud is analyzed.

KEYWORDS
BIM
Parametric Modeling
Point cloud

Foreword

This master thesis is the final thesis for the 2-year master's degree in the "Structural Engineering and Building Technology" program at OsloMet – Oslo Metropolitan University. The thesis is written in collaboration with Oslo Metropolitan University in the spring of 2022.

BIM is a topic that is constantly growing and evolving every day and is a topic that was introduced early in our studies. Through our studies we were introduced to various BIM-tools, applications, and the potentials they had.

Today, there is a demand for a smarter building industry and BIM and related processes can be the solution. Take parametric design as an example. Parametric design is more in focus than it has been before, and the use of this technique can be seen in various building designs and constructions. Parametric design can cover any field of design, everything from engineering or architecture to interior design.

Through this thesis we have ascertained extensive knowledge and expertise in the field of BIM, parametric design, 3D laser scanning, programing and not the least various applications and tools. This thesis has given us insight into these fields and how they can be a vital part in the future of the AEC-industry.

We would like to thank our supervisors Prof. Daguang Han and Prof. Gro Markeset at Oslo Metropolitan University for the support and guidance they have given us during this thesis. We would also like to thank Prof Daguang Han's research team in China, especially Yu Zhang and Kaixin Hu. Their knowledge and guidance have been vital for the completion of this thesis.

Abstract

Bridges have a crucial and huge role in the safety of the public. An important aspect in maintaining the safety of a bridge structure is to have access to a BIM-model that can be used in analysis and facility management. Long-span spatial curved bridge structures with circular pipe components are widely used in engineering. Measuring the overall alignment of long-span spatial curved structures with circular pipe components quickly and accurately is an important aspect that increase the effectiveness of creating digital twins. This thesis has an approach to find a method to measure this overall alignment in an accurate and quick way.

The method in this thesis is applied to a real case study which is a pedestrian bridge located in Oslo, Norway. The bridge consists of curved steel members at top chord and circular pipe components as struts.

The method implements terrestrial 3D laser scanning of the case bridge for data collection. The data obtained from the scans has been processed and imported into MATLAB to extract geometry information of the components automatically by applying algorithms. The information and parameters extracted from MATLAB are then imported into Dynamo to generate a parametric driven BIM-model of the existing bridge. The model is compared with the point cloud of the bridge to analyze the deviation between the point cloud and the parametric driven BIM-model.

The results obtained from various operations have a good level of accuracy, but there are still room for adjustments in the method that can improve the accuracy of the results. The case study examination in this thesis was however successful, and it can be concluded that the method applied in this thesis can help to measure the overall alignment of long-span curved structures with circular pipe components quickly and accurately.

Content

Terminology

AEC	Architecture Engineering and Construction
BIM	Building Information Model/Modeling
CAD	Computer-aided design
NURBS	Non-Uniform Rational B-Splines
VDC	Virtual Design Construction

List of Figures

List of Tables

1 Introduction

1.1 Background

Bridges have a crucial and huge role in the safety of the public. Hence, it is important to inspect bridges in a proper way to provide the confidence to the owner of the bridge and the public. The inspection work contains of controlling the condition of the bridge, updating the information, evaluate and analyzing the information found, and reporting the situation. [1]

An accurate model of an existing structure will help engineers analyze and inspect a structure more effectively. Such a model gives a better understanding of the structure to the owners as well by providing a real picture of an as-built structure. Such a preliminary digital twin model can be used for the follow-up structure monitoring work. The subsequent structural monitoring includes digital preassembly, on-site assembly, structural attitude monitoring and analysis, overall stress performance analysis and safety evaluation of the structure after welding, etc.

Long-span spatial curved structures with circular pipe components are widely used in engineering, such as the main skeleton of some large steel structure buildings, steel pipe arch ribs, cable-stayed bridge cables and suspension bridge cables. One of the main problems in the building industry regarding such structures is measuring the overall alignment of this type of structure quickly and accurately.

The digitalization in building industry and capability of BIM can be utilized to find a solution to the mentioned problem. In this thesis, it is attempted to solve this problem by using the theory and method of digital twins and the means of BIM and three-dimensional laser scanning. The starting point is to use three-dimensional laser scanning to obtain the three-dimensional point cloud model of a long-span spatial curved structure with circular pipe components, then use algorithms to extract the characteristics of line shape and radius, and finally use the parametric BIM modeling method for the construction of the digital twin model.

1.2 Research Question

It is a difficult problem to accurately obtain the alignment of long-span spatial curved structure with circular pipe components. The traditional method is to use the total station to measure the reflector attached to the lower edge of the circular pipe structure, and further fit the alignment of these limited points. Due to the importance of obtaining an accurate alignment of such structures efficiently, it is attempted to answer to the following research question:

"How can the overall alignment of long-span spatial curved structures with circular pipe components be measured quickly and accurately?"

1.3 Limitations

The technical route of this thesis is based on a real case study. The methods applied in this thesis have met some challenges which may have affected the quality of the results. In this subchapter, some limitations regarding the work done in the thesis are described briefly.

The first step of the method used in this thesis is to find a steel bridge with curved structure and circular pipe components to scan. Finding such a case took a long time due to the unavailability of such structure near the place authors of the thesis live. The most suitable case is a structure that is curved in only one plane. However, the case found for this thesis is a pedestrian bridge that is curved in two planes and directions as shown in figure 1.1. The details about the effect of this limitation are described in chapter 5 and 6.

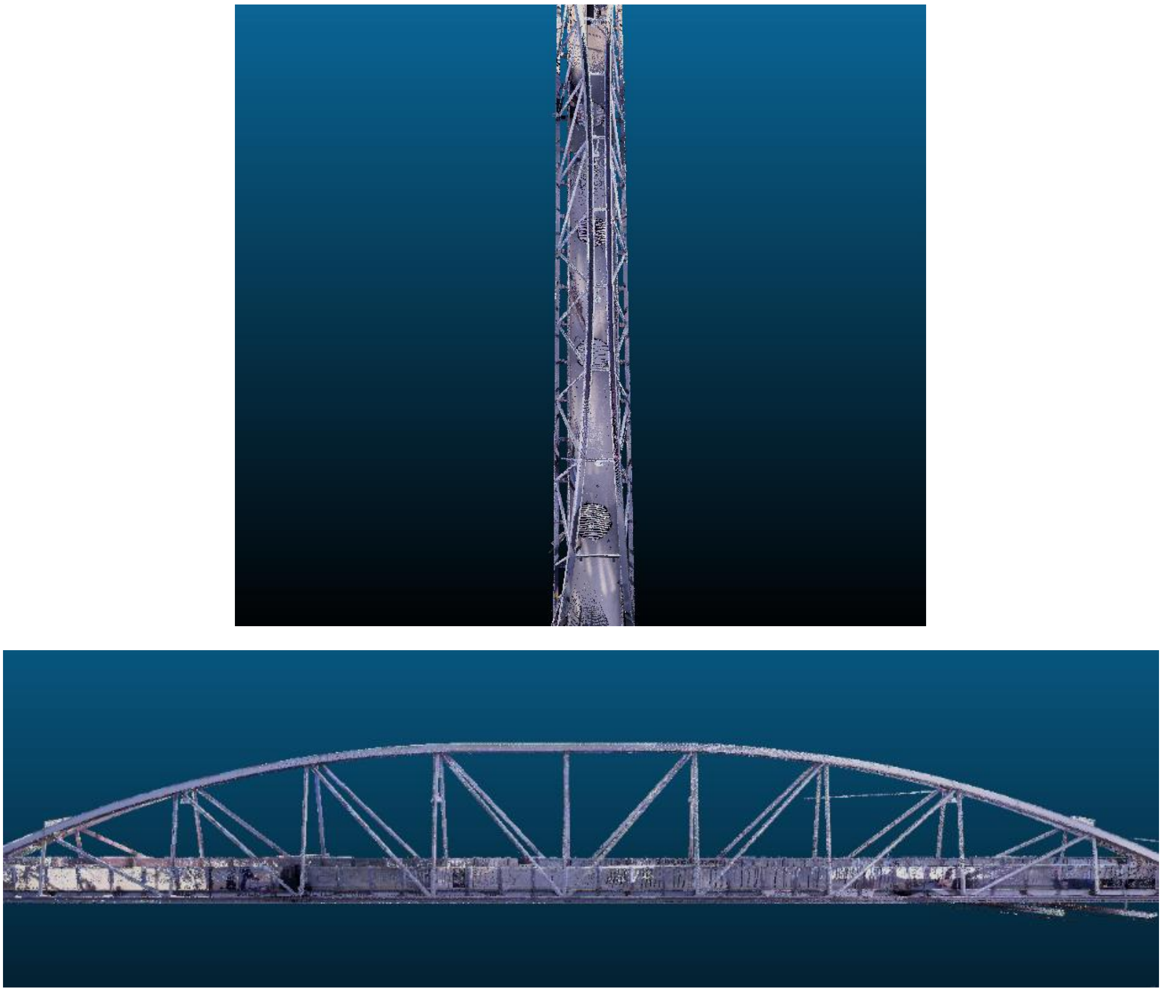

Figure 1.1: The bridge that is curved in two directions

The 3D laser scanner used in this thesis is owned by Oslo Metropolitan university. The availability of the scanner and the possibility to borrow it can also be considered as a limitation. Another limitation regarding the scanning of the bridge is the availability of the free space inside and around the bridge so that the scanner can be placed freely to get best results. High amount of traffic from pedestrians on the bridge and vehicles under it has limited the ability to place the scanner freely.

Many other methods described in chapter 4 are based on using software tools to deal with the case. Some software program’s license is available through the university. However, the limited license for some programs such as Geomagic Control X have led to limitations regarding the work with the related data to get most accurate results. Another issue due to use of software tools is the computing power. Software tools such as MATLAB, Geomagic Control X and Dynamo require high computing power to deal with huge algorithms, adjustments, calculations and runs.

The majority of the work done in this thesis is based on the knowledge about the various algorithms and software tools. The knowledge in this case can be defined as both pre-knowledge about the data and knowledge about the way software tools work to obtain reliable results. Reaching such level of knowledge in short period of time for such a complex case can be considered as limitation. The methods used in this thesis are of an art that do not necessarily match the background and pre-knowledge that the authors have had.

However, the authors and supervisors have managed to deal with most of the limitations and the details about the effect of limitations are described in chapter 6.

2 Theory

2.1 BIM

BIM stands for building information modeling, but also known as building information model. BIM is a digital representation of a building's functional and physical properties. These features are represented as a digital 3D model and can be used as a visual aid throughout all phases of a project. BIM is seen as both a process and a data model.

In addition to BIM being a visual aid, it can be used as an information model where each object in the model contains detailed properties and specifications associated with the building. The objects can also contain documentation about management, operation and maintenance, which makes the model useful even after the project has been completed and constructed. The ability of BIM to collect all information about a building in one place and its contribution to a well-flowing interdisciplinary project collaboration makes the process for users easier. This information is available throughout the life of the building. The different actors involved in a project can work in one and the same model. This contributes to better communication, and the outcome is that errors and shortcomings can be detected earlier, which in turn leads to saving time and money.[2]

Figure 2.1: Illustration of Building Information Modeling[3]

2.1.1 History of BIM

The idea behind 3D visualization and modeling of buildings started as early as the 1980s. Architects began using PC-based CAD during this period [4], which was preferred over computer programs for building design. The progress of BIM met the challenge of high cost of powerful computers that were able to handle these programs. The development of BIM and BIM software proceeded parallel with the technological development of computers. The BIM evolution continued with the introduction of object-oriented CAD in the early 1990s and formed the basis for BIM [5].

2.1.2 Why BIM?

The demand for a smarter building industry due to the increase in the population is a fact that cannot be neglected. It will be up to future civil engineers and the entire AEC community to look for smarter, more efficient solutions to build, project and design.

BIM gives the opportunity for design and engineering managers to work more efficiently. In addition, with the help of BIM, one can also acquire data that is created during the process for the benefit of operation and maintenance. This is one of the reasons why BIM users are increasing more and more over the years.

Through the various phases of the construction process, one can expect many benefits from using BIM. Here are some benefits users can get from using BIM models and related processes[5]:

- Always updated BIM model by maintaining the accuracy of the as-built model when changes and additions are made to the building during the life of the building.
- By exporting and integrating current as-built models and equipment information to the systems to be used over the structure's life cycle, maintenance and structure management are optimized.
- Provides a basis for VDC and LEAN construction building practices. VDC is an execution strategy that is interdisciplinary. It aims to streamline planning and execution of projects, as well as increase quality and value for the customer by combining BIM with Lean thinking and practice. Lean construction also has as its main task to create value for the customer, for example through integrated project delivery and targeted design.

- Shorten the project planning, from pre-design to finished product, by coordinating and prefabricating parts of the design using the building models with reduced working hours.
- One can get more reliable cost estimates early and good interaction with the project group by using the contractor's and / or the architect's BIM models. This leads to reduced financial risk associated with the project.
- Through BIM-based lighting and energy design and analyzes, one can improve the quality of the building and get a more sustainable building. Building quality is a characteristic of a building that tells how well the building performs its functions.

2.2 Parametric Design

There are many ways of understanding design, and this is a result of the technology that keeps developing every day. Parametrical design is one way of understanding design. Parametrical design can be described as a system of parameters, variables and restrictions that with the application of specialized software can create flexible objects. “Parametric” is a term that stems from mathematics and introduces the use of specific variables or parameters. The specific variables or parameters, that can be edited, are used to manipulate or alter the end result of an equation or system [6]. A cube can be used as an example, where the variables that can be introduced are the cubes length, width, height and its edges. These variables can be used to adjust the dimension of the cube, and parametric design involves the change of these variables in order to acquire a completely different object, which is achieved with the help of algorithms [7].

2.2.1 History of Parametric Design

Parametric design is not a new and foreign concept. In the past buildings have been designed and constructed in relationship to a variety of elements such as technology, culture, climate, use, mood, setting and character [8].

Antoni Gaudí is a Spanish architect who used parametric design and his famous work is found throughout Barcelona. He started using parametric techniques at the end of the 19th century and his upside-down model of churches is one of the earliest examples of parametric design. In the design for the “Church of Colònia Güell”, Antoni Gaudí created an upside-down model of strings weighed down with birdshot in order to create complex arches and vaulted ceilings.

With this method he could manipulate and alter the shape of each arch and see the outcome of the change and in addition how the change affected the arches connected to it. With a mirror placed at the bottom of the model he could see what the church would look like right side up.

This analogue method of Antoni Gaudí consists of all the characteristics of a parametric model, which consists of:

- Input parameters
- Equation
- Output

The anchor point location, the string length and the birdshot weight are the independent input parameters. The outcomes of the model are the vertex locations of the points on the strings and the outcomes are derived by functions with respect to gravity and Newton's Law of Motion.

Different versions of Gaudí's model could be obtained, with 100% certainty that the structure would stand in pure compression, by modifying some parameters of the model. There was no need for Gaudí to calculate the results of the parametric equations manually, since the shape of the catenary curves could automatically be derived through the force of gravity acting on the strings [6].

Figure 2.2: An upside-down model of the Church of Colònia Güell [9].

2.2.2 Why Parametric Design?

Today, parametric design is more in focus than it has been before, and the use of this technique can be seen in many building designs and constructions. Parametric design can cover any field of design, everything from engineering or architecture to interior design, and even fashion.

With this technique of design geometric shapes are left behind for harmonic, continuous, organic and fluid structures. Unique and futuristic spaces are created with its application to industrial and architectural design.

In the past, data was presented in order to accomplish a goal without a great deal of freedom and projects were seen and executed as something rigid. Parametric design has today enabled projects to become a process with the possibility to alter and modify constraints and variables almost instantly, which generates adaptable and versatile designs. The time and money saved from this makes parametric design economical and time efficient.

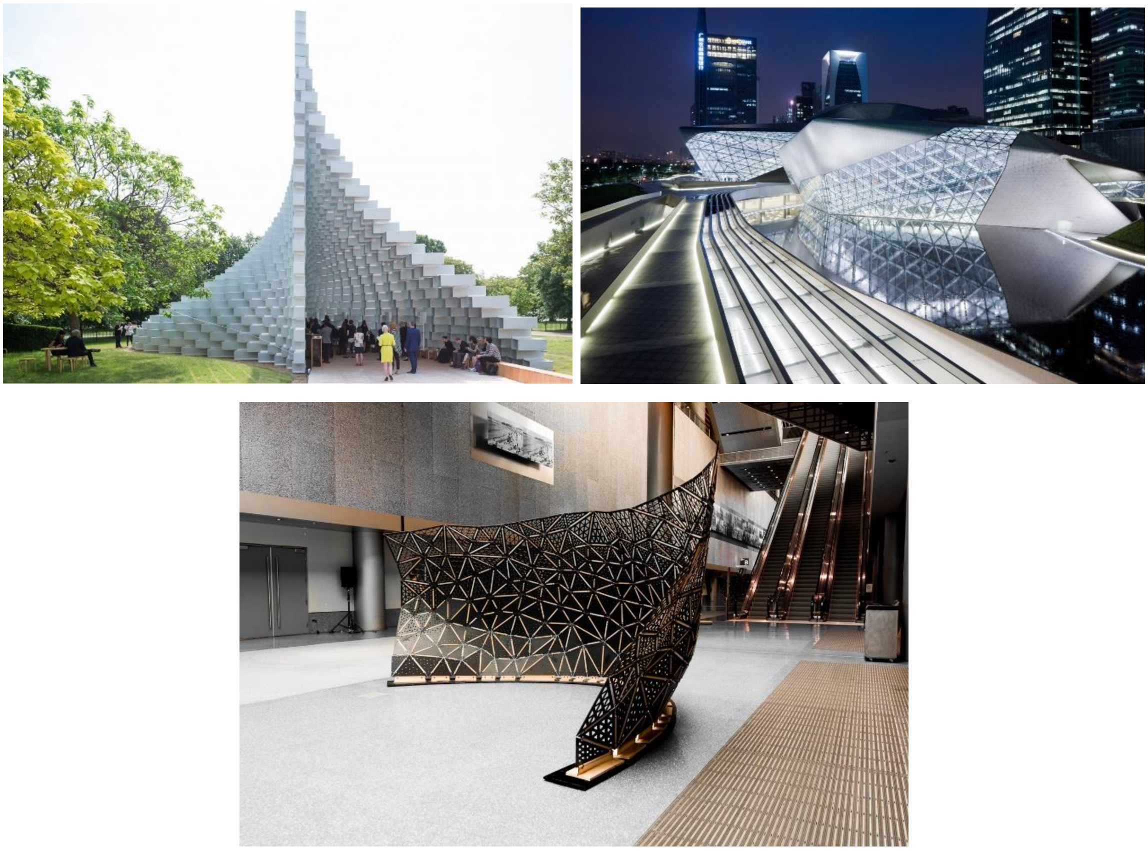

Figure 2.3: Illustrations of unique and futuristic structures [10].

Parametrical design offers a great deal of advantages, such as [7]:

- **Significant freedom.** One can create irregular and fluid shapes, such as organic shapes which are the most common and noticeable and inspired by nature. Organic shapes transmit dynamism and movement and are characterized by their unpredictable and irregular contours. They provide continuity and fluidity by adapting to the generic spaces.
- **Uniqueness.** The parametric software produces a variety of possibilities when any variable lines, points, volumes or planes are altered and changed. This results in unique and newfound designs.
- **Adaptable and flexible.** The design can be adapted to the needs of people or places due to the flexibility. Parametric design enables e.g., corners, shelves, columns and stairs to be integrated into spaces in an organic and natural way.
- **Aesthetics.** Without abandoning functional and practical advantages of spaces, parametric design gives the spaces a significant level of distinction and character.
- **Unlimited use.** Parametric design has no limits besides the imagination and can be used in commercial, residential and other types of buildings and structures, and everything from wood to metal as material.

2.3 Digital Twin

Digital Twin is a virtual representation of a physical object or process. The digital twin is updated through real-time data and uses machine learning and simulations to reflect the physical object or process exactly. By applying sufficient and accurate data, the virtual model can be utilized to perform study performance issues, simulations, generate potential improvements, and help overall decision-making. The purpose of all of this is to produce valuable insights that can in a later stage be applied to the initial physical object or process [11].

The concept of digital twin is not something new, and the idea of studying an object or process through an exact virtual model can be seen as early as the 1960s. NASA paved the way when it comes to the use of the concept digital twin when they exactly replicated each voyaging spacecraft in the 1960s. The virtual model was used for simulation purposes and study.

Figure 2.4: Illustration of digital twin [12].

Many benefits are expected with the use of digital twin technology. It provides plenty of data on performance outcomes and presents more effective design and research of products. The information and data that has been gathered are assessed by the companies in order to get insight and an overview on the things that need be improved before the production. Even after production the digital twin can be utilized to monitor and mirror production systems, with the goal of efficiency throughout the entire manufacturing process. When it comes to the end of a products lifecycle, the digital twin can help determine which product materials that can be harvested and help the manufacturers decide what to do with these products.

2.4 Software Programs

2.4.1 Revit

Revit is a software used to produce consistent, coordinated and model-based design and documentation. Revit has several benefits, such as 3D visualization that allows one to see what a building looks like before it is built. Other advantages are automatic updating of 3D views, sections, slopes and floor plans, and ability to perform a solar study of the plot to be built on. In Revit, participants in a project can work with the work sharing tool that allows storage and sharing of information in the same project. Projects can be exported or imported in various file formats as Revit is compatible with most file formats such as the IFC file format.[13]

Figure 2.5: Capability of Revit and BIM [14]

2.4.2 Recap

Recap stands for "Reality Capture" and is a software for processing point clouds. With Recap, you can directly open point-cloud files and filter data that are not necessary. Hence, the amount of data and storage space will be reduced. One can use Recap's point file to clean up a scan of an existing building, then import it into other software such as Navisworks and Revit. Using Recap, one can create 3D models from imported images and laser scans. Then one can produce point clouds that are ready for further processing through CAD and BIM tools. In addition, through the properties of Recap you can edit, measure and visualize the point cloud.[15]

2.4.3 Dynamo

Dynamo is software tool that can be used in either a stand-alone mode or as a plug-in for other software programs and tools such as Revit or Maya. Dynamo has the goal of being accessible to both programmers and non-programmers and is a visual programming tool that provides the users the ability to script using various textual programming languages, define custom pieces of logic, called custom nodes, and visually script behavior [16].

Dynamo does not require the user to have any previous experience with programming. It has an easy-to-use simple scripting interface and offers open-source graphical programming. Dynamo extends the capabilities and works as a supplement of Revit, which make way for BIM-connected computational design. Using surface and solid geometry the users can visualize and analyze projects. In addition, by using graphical logic, geometric problems can be visualized, a test of the behavior of the design model can be conducted and complex design challenges can be addressed.

Other benefits of using Dynamo are [17]:

- Communication with Excel, Robot, Navisworks, SAP2000, Google Sheets, Revit, Rhinoceros.
- Iteration of designs and rapid testing.
- Using simple logic, data and analysis generate sophisticated designs.
- Risk studies.
- Automation of repetitive tasks.
- Solutions to geometric problems that are complicated

2.4.4 MATLAB

MATLAB is a programming and numeric computing program used by engineers and designers to analyze and process data and make models. The program is developed by MathWorks and uses a programming language that demonstrates matrix and array mathematics directly. Some characteristics of MATLAB are:

- Ability to inspect, model and analyze data
- Ability to visualize data
- Programming with creation of scripts and functions
- Ability to connect to hardware
- External language interfaces
- Possibility for designers to share the program and run it in a cloud from MathWorks to public clouds
- Development of algorithms
- Running huge and complex computations with simulation

Some significant areas of application of MATLAB are machine learning and deep learning, control systems, robotics, image and video processing, enterprise and IT systems, data science and test and measurement.[18]

2.4.5 CloudCompare

CloudCompare is originally known as 3D point cloud and mesh processing software. Lately, CloudCompare has been used as a generic point cloud processing software. The processing of point cloud contains for instance cloud registration, resampling, distance computations, statistics, display enhancement and interactive or automatic segmentation.[19]

The software can perform unlimited scalar fields per point cloud based on algorithms such as smoothing, statistics and gradient evaluation. The ability of software to segment 3D entities, picking points and extraction of 3D data from components in a point cloud has made the software popular in processing point clouds obtained from 3D laser scanning.

2.4.6 Geomagic Control X

Geomagic Control X is a software for dimensional inspection and 3D quality control. The software allows the user to capture and process data that has been obtained from 3D scanners and other devices to understand, communicate and measure inspection results in order to ensure quality. With Geomagic Control X the users can gain deeper insights and optimize manufacturing processes.[20]

The main reasons to why Geomagic Control X is so popular and well-liked software program are [20]:

- It is an easy to learn software program. Anyone can get results independently of their level of knowledge and skills
- It is a fast and easy to use software program. Huge data sets can be handled easily and quickly.
- It provides all the tools the user needs for a professional-level inspection.
- It is a software program that is compatible with all 3D scanners. The user can work with any 3D scanner or PCMM arm.

2.5 Laser Scanning and Point Cloud

Laser scanning is a method of collecting 3D point data. Large amounts of point data, so-called point cloud, are points that are defined in three dimensions. The method is used to create virtual models that can be used in software for visualizing objects and areas.

When performing a scan with a laser scanner, the laser scanner detects a large number of data points which are reflected back from the surface that is scanned. These can be surfaces such as walls, windows, steel structures, ducts, etc. The data points together form what is called a "point cloud".

The point cloud captured by the laser scanner can be considered as a precise sketch of an as-built of an area or object. The "sketches" of objects and areas can be made by the data points covering the surfaces of the objects and areas, and the more points the more accurate the sketch becomes.

Since data points cannot be retrieved from surfaces the laser scanner does not see, it is necessary to scan from different locations towards the area or object being scanned to get a complete overview. By using, for example, Autodesk's program, Recap, you can easily merge the different point clouds so that you can see the entire area or object. By merging the point clouds, you improve the quality and reduce the amount of data. Merging point clouds requires good structure and planning.

There are several ways of 3D laser scanning available today such as aerial laser scanning, mobile laser scanning and terrestrial laser scanning.

2.5.1 Aerial Laser Scanning

Aerial Laser Scanning is a method when a laser scanner is sticked to and carried by a helicopter or a drone. This method is largely used to make a 3D point cloud of a terrain. This method measures the surface of the earth by pulses of light and their reflections. These pulses are emitted from the scanner. The point cloud in this case can be used to make a model of the construction site. Since the distance that the scanning is performed is long because of the height, the resolution and accuracy of the point clouds in some cases may become low. However, this technology seems to be cheap, and is widely used. Another advantage of this method is the ability to gain data from features under a forest canopy. Other use areas are archeology, urban planning and geology.[21]

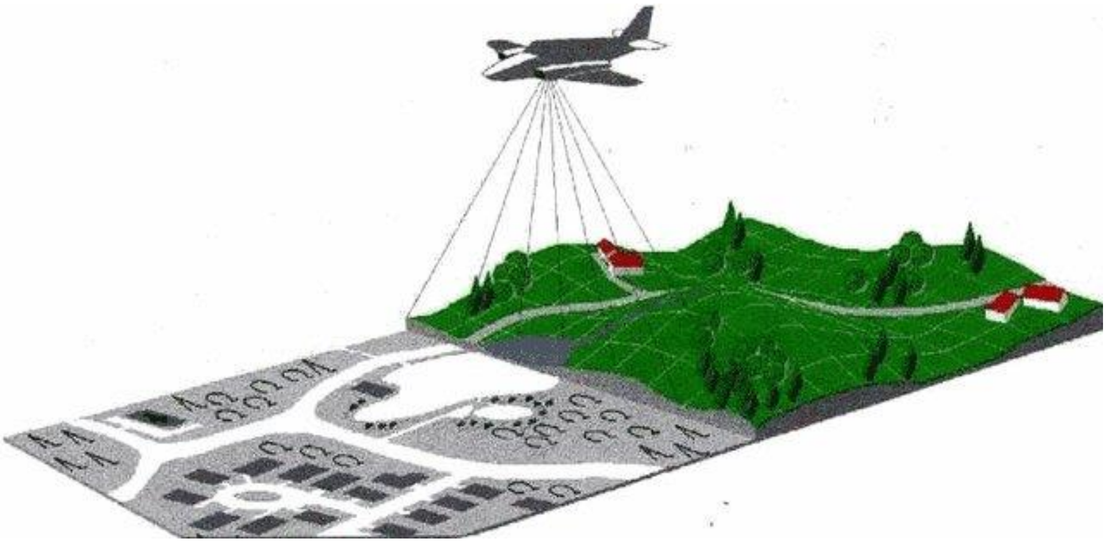

Figure 2.6: Aerial Laser Scanning[22]

2.5.2 Mobile Laser Scanning

The laser scanner in this method is mounted on a moving object. These objects can be cars, trains, bikes, boats, helicopters and even humans. The main area of utilizing this method is to study landscapes, urban areas, buildings, and roadways for road quality control. This technology has become more popular than the traditional way of surveying due to the time and budget saving capability. The daily traffic will also not be interrupted as much as it does in traditional surveying methods. The method is suitable in time sensitive projects and can be used in mapping projects such as Google Maps and Street View. The largest characteristic of mobile laser scanning is their capability to scan many miles of land in a very short time.[23]

Figure 2.7: Mobile Laser Scanner mounted on a car[24]

2.5.3 Terrestrial Laser Scanning

Terrestrial Laser Scanning refers to a scanning done from a ground based stationary scanner. The scanner is normally mounted on a three-legged stand with a fixed position. The scanner rotates then horizontally and vertically, and measures distances by emitting laser beams. The principle of measurement for such scanners can be divided into two. The first type is called time of flight which calculates the distance by emitting and receiving a light pulse. The time for the round-trip is then captured, and the distance is obtained from the velocity of the light pulse and the time for round-trip. The second type is called amplitude-modulated continuous-wave. The principle of measurement for this type is that the scanner emits amplitude modulated continuous wave and calculates the distance by recording the phase difference between emitted and reflected waves. The first type is normally used for long range scanning. The quality of the scanning normally becomes lower due to the long distance. The amplitude-modulated continuous-wave scanners has a higher level of accuracy and are normally used for close range scanning. [25]

Terrestrial Laser Scanning is widely used to generate a 3D-scan of existing buildings and capture reliable data in real time of as-built structures due to its high accuracy.

Figure 2.8: Terrestrial Laser Scanning[26]

Since a Terrestrial Laser Scanner has a fixed position during the scanning, it leads to blockage or occlusion. It is capable to scan the surface of objects around, but it cannot capture and make a complete 3D view of objects. To solve this problem, several scans need to be done from different angles. The scans made can then be merged to get a full 3D-view of a 3D object.

2.5.4 Point Cloud Merging and Processing

It is common for two or more point clouds to merge at points which are common for the point clouds. It is suitable to choose points that are easy to find in the clouds. «Targets» can be used for this purpose. Targets are reference points that are placed around the area or object to be scanned and can be as simple as a paper printout. It is important that these targets are scattered both horizontally and vertically. These targets become common points for the software to recognize when the point clouds are to be merged. Another example for the targets is target spheres. The surface of the spheres is made of various material with different reflectivity.

Figure 2.9: Example of a checkerboard target and target spheres [27, 28]

With good planning and well-thought-out locations of the targets, the time for merging the point clouds will be significantly reduced. When scanning from different locations, there should be at least one common target in the different scans.

Accuracy of a point cloud obtained from a scanning is highly dependent on the type of laser scanner and the way it is calibrated. Various laser scanners have different quality in scanning, different range and settings for the calibration.

Often the point cloud includes unnecessary data. Therefore, the point cloud must be cleaned before proceeding to the modeling. Some typical sources of errors are:

- Poor lighting that causes objects to reflect the light beam poorly or the light beam to hit an object unevenly
- Distance to the scanner that can cause the light beam to split in two when it hits an edge, and reflects from two different distances
- Dust and steam that can refract the light beam and reflect it
- Reflection from unwanted objects behind an object one has chosen to scan
- Multiple reflections of the same light beam

With this, one can use software to manually delete points that are in the wrong place, or that are not relevant. This process is called noise reduction.

After importing the point cloud into a point cloud processing program such as Recap, the point cloud can be analyzed, modified and manipulated to suit the user's needs. Furthermore, data and information can be converted to a CAD model by importing further into a CAD-based modeling program, for example Revit, where data and information can be turned into 3D shapes and surfaces such as walls, floors, terrain, etc. [29]

2.6 Algorithms

Algorithm is defined as *“a process or set of rules to be followed in calculations or other problem-solving operations”*. In other words, algorithms introduce a set of instructions and rules that define how a work is implemented step-by-step in order to achieve the expected results. A simple example is for instance the cooking of a new recipe. The recipe has clear instructions and steps that needs to be followed and executed one by one and in the given order and the result is a perfectly cooked dish.

Algorithms are language-independent which means that the instructions or the set of rules of the algorithm can be implemented in any language and the result and output will be the same. Not every written instruction or set of rules are considered as an algorithm. Algorithms have their own characteristics such as:

- They have clear steps in all aspects which leads to only one meaning.
- They have well-defined inputs and outputs.
- They must be practical, generic, and simple in order to achieve it with the resources available.
- They must be language-independent.
- They must not end up in for example an infinite loop and must be finite.

The advantages of algorithms are greater than the disadvantages. Algorithms are easy to understand and gives a step for step solution to a given problem or obstacle. An algorithm breaks down the given problem into smaller steps and makes it easier for the programmer to convert and develop it further into an actual program. On the other hand, writing algorithms can be time-consuming and tricky to handle. Branching is decisions on what actions to take and looping is the decision of how many times to take the certain action[30]. This is something that is difficult to show in algorithms.[31]

2.6.1 Unsupervised Machine Learning

Unsupervised learning is known as unsupervised machine learning and is based on unlabeled dataset. These unlabeled data can be analyzed and clustered with help of machine learning algorithms. The main characteristic of unsupervised learning is to uncover and locate patterns and data groupings with help of algorithms. In other words, it discovers the similarities and differences in information.

The main functions of unsupervised machine learning are clustering, association, and dimensionality reduction. Clustering is a method that groups unlabeled data based on similarities and differences. The patterns obtained from clustering algorithm are based on raw and unclassified data. The association rule refers to finding relationship between variables in a dataset. The dimensionality reduction refers to reduction of dataset for a better performance of an algorithm. The amount of data available affects the accuracy of the results. The results become more accurate when there are more data available. However, performing an algorithm and visualizing dataset can in some cases become challenging. Hence, it is important to reduce the number of dataset input by using dimensionality reduction technique in a huge dataset. [32]

2.6.2 Euclidean Clustering Algorithm

One of the algorithms that can be used for segmentation of structures in a point cloud is Euclidean clustering algorithm. The algorithm is based on recognition of the points that are close to each other. Therefore, one should set a threshold for the closeness in form of distance. The algorithm then recognizes that the points within the threshold belong to a specific and described cluster. There is also possible to set a minimum and maximum number of points that each cluster can contain. The algorithm can also be used in denoising components by limiting number of points that can belong to a cluster. [33]

The Euclidean clustering algorithm, also known as the K-means algorithm, uses centroid points to establish clusters, where the “k” is the number of clusters. The clusters can be measured using intercluster and intracluster distance. The distance between data points in different clusters is the intercluster distance. The distance between data points inside the cluster is the intracluster distance. The average of all data points in a cluster is known as a centroid point. Each centroid point can be assigned to a cluster by assessing the Euclidean distance between each point in the dataset iteratively. As the process is carried out, the centroid points start by being random in the beginning and will later change each time as the process continues. In cluster analysis, the K-means algorithm is a common algorithm that is mainly used for scalar data. [34]

2.6.3 Polynomial Curve Fitting

Curve fitting can be defined as a way to present or model data by applying the best possible function or curve along the range of the data. When the curve or function picks up the course of the data, it allows for prediction of the behavior of the data series.

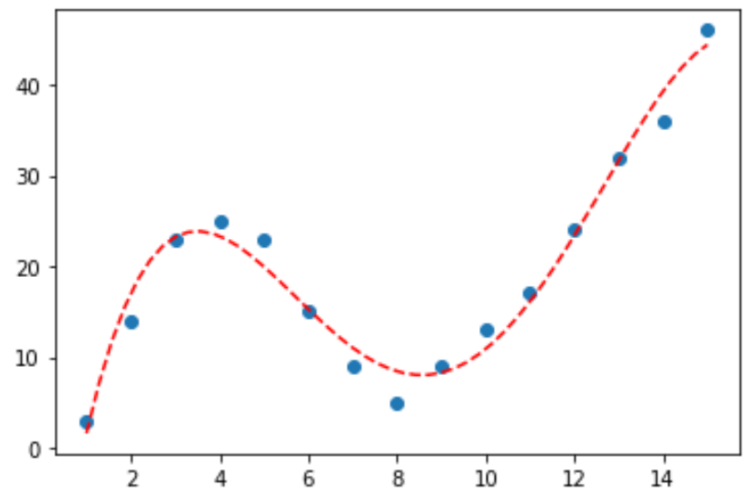

Figure 2.10: Illustrative example of curve fitting [35]

When it comes to polynomial curve fitting the data is fit to a graph of a polynomial function. One of most common methods of obtaining the polynomial of a given degree is the least squares method. [36]

The least squares method is a branch of regression analysis. This method is used to find the best fit for a set of data points. The points in the data set each represents the relationship between an unknown dependent variable and a known independent variable. The best fit for the data points is found by minimizing the sum of residuals of points of the plotted curve. The least squares method can be utilized to predict the dependent variables behavior. In figure 2.11 an example of polynomial curve fitting is shown. [37]

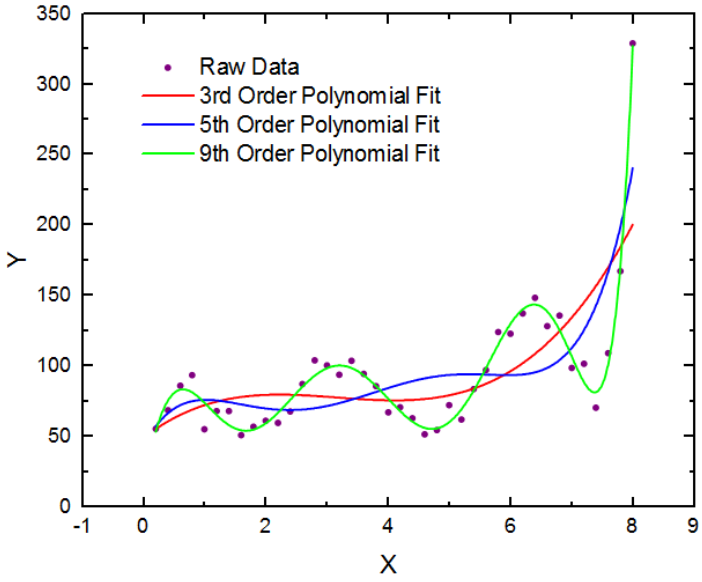

Figure 2.11: Illustrative example of polynomial curve fitting with different polynomial degrees [38]

2.6.4 Sliding Window Technique

The sliding window technique is a well-known and common technique used by programmers and is utilized in various algorithms. The functionality of this technique is just as simple as the name. When using this technique, a window is created around a set of data. The window has the ability to slide so that different portions of the data can be captured. In figure 2.12, an example of the sliding window technique is shown. In this example a random set of data and a sliding window of size 3 is presented. The start point of the window is set at the first number in the data set and the end point is at the third number in the data set since the size of the window is 3. The window captures the first 3 numbers in the data set before it slides to the right with one step and captures 3 more numbers. This process is carried on until the window reaches the last 3 numbers in the data set. [39]

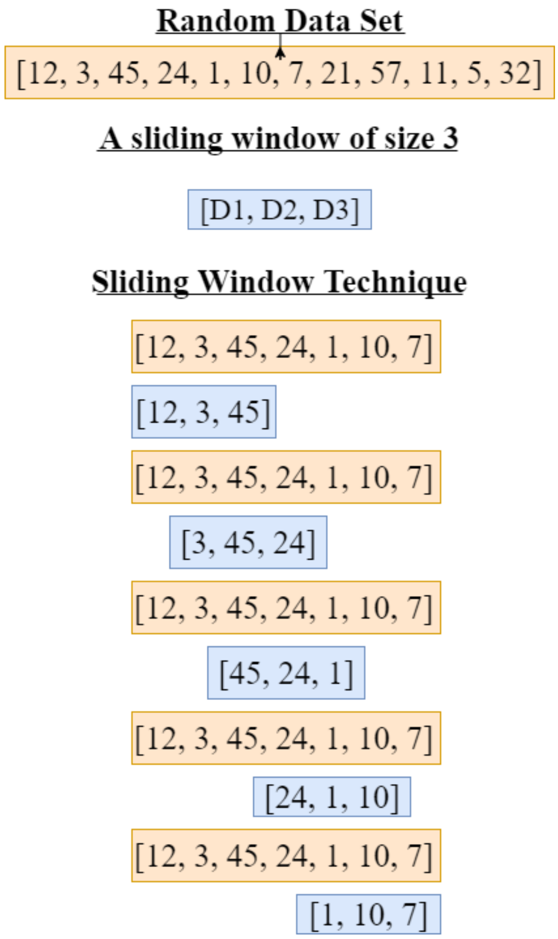

Figure 2.12: Example of Sliding Window Technique

3 Case study

The research in this thesis considers finding a way or ways of obtaining the overall alignment of long-span spatial curved structure with circular pipe components accurately. Based on the technical route of the thesis, the first step is to scan such a structure by 3D-laserscanner. Finding a structure with requirements stated in the research question will make the results of the thesis more reliable.

The authors of the thesis have contacted various companies to find a suitable case. Finding a proper case has been quite challenging. The case that is used in this thesis is a pedestrian bridge made of steel components. The pedestrian bridge is in the center of Oslo city in Norway. The owner of the bridge is Norwegian Public Roads Administration. After contacting the authorities, the permission for scanning the bridge was conducted.

The bridge is a truss which consists of curved I-shaped steel structure as top chord, circular pipe steel diagonals, straight I-shaped steel bottom chord, I-shaped beams and slab as shown in figure 3.1.

Figure 3.1: Picture of the pedestrian bridge in Oslo

The main components of the bridge that are under research in this thesis are top chord and diagonal struts. The design data and model of the bridge are not available during the research, and all data gathered in this thesis are data that are obtained from 3D-laser scanning done by the authors.

The challenges regarding the case are high traffic from pedestrians on the bridge and vehicles under it, since the bridge is located next to Oslo central station which is crowded all the time. This affects the process of denoising the point cloud and ability to place the scanner freely around the bridge to get an accurate point cloud of the bridge.

4 Method and Technical Route of this Thesis

Various factors in design, construction and operation phase can affect the deformation of a steel structure. The built structure has in most cases some deviations from the designed model. This deviation may come from the manufacturing stage, constructing stage, or miscommunication in designing phase of a structure. However, it is important to have access to a 3D-model of the real as-built structure for management, analysis, and maintenance of the structure. This thesis' content is about making a Digital Twin of a pedestrian bridge with enough accuracy of overall alignment effectively.

In order to detect the eventual bridge deformation caused by factors under design, construction and operation phases, and because of the lack of accuracy and efficiency of traditional detecting methods, the 3D-laser scanning is chosen to collect data. The hardware used for this purpose is the 3D-laser scanner Topcon GLS 2000. The scanner is lightweight which can capture a full 360˚ scan including images. Some key features of the scanner are automatic temperature adjustment, precise scan technology, and multiple lens array system[40].

Figure 4.1: Topcon GLS 2000 laser scanner [41]

The traditional method is to use the total station to measure the reflector attached to the lower edge of the circular tube structure, and further fit the alignment of these limited points.

The 3D-scanning of the bridge requires good planning to get an accurate point cloud. By studying the geometry and the position of the pedestrian bridge, it will be easier to make a plan for placement of the scanner during various scanning's. After the data collection, the data processing stage begins. This stage includes point cloud registration, denoising, compression, etc.

The scope of this study is to build a parametric 3D-model of an as-build structure with high accuracy and efficiency. Dynamo in Revit is used for the parametric modeling of the structure. The software builds a model based on parameters and variables. The point cloud gathered from the scanning's is only a huge number of points captured by the 3D-scanner. Hence, the point cloud needs to be processed with help of algorithms before the input into Dynamo.

Using 3D laser scanning to obtain the 3D point cloud model of a long-span spatial curved structure with circular pipe components needs to use relevant point cloud processing algorithms. Normally, the point cloud processing algorithm includes station matching, denoising, and automatic extraction of curve alignment and circular component's radius. Among them, the first two algorithms are existing algorithms, which exist in commercial software tools, and the third algorithm needs to be programmed in MATLAB. After this post-processing of the point cloud, the curve shape and circular pipe radius that are extracted will be parameterized in BIM software. The deviation between the established solid circular pipe and the collected point cloud can then be analyzed, and the accuracy and efficiency of the process can be measured.

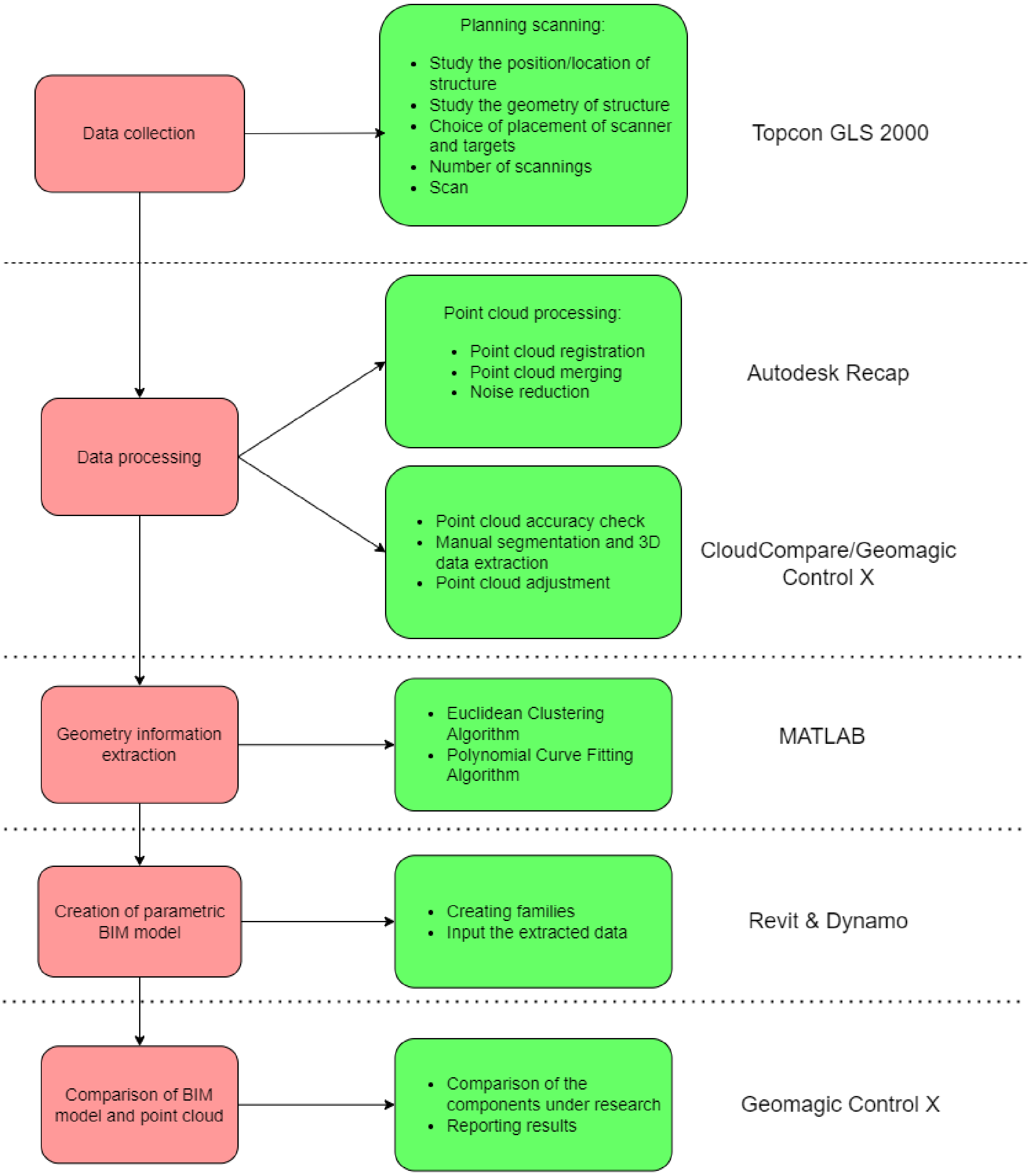

Figure 4.2: The technical route of the thesis

5 Implementation and Results

This chapter contains the description of implementation of the steps in technical route of the thesis. The results gained from each part will be shown in this chapter as well.

5.1 Laser Scanning and Point Cloud

The first step in the technical route of this thesis is data collection. 3D laser scanning is used for this purpose. The laser scanner used in this case is Topcon GLS 2000 owned by Oslo Metropolitan University. Various 3D laser scanners have different settings. Hence, it is important to study the instruction manual from the producer before starting the scanning. Some important settings to mention are shown in figure 5.1.

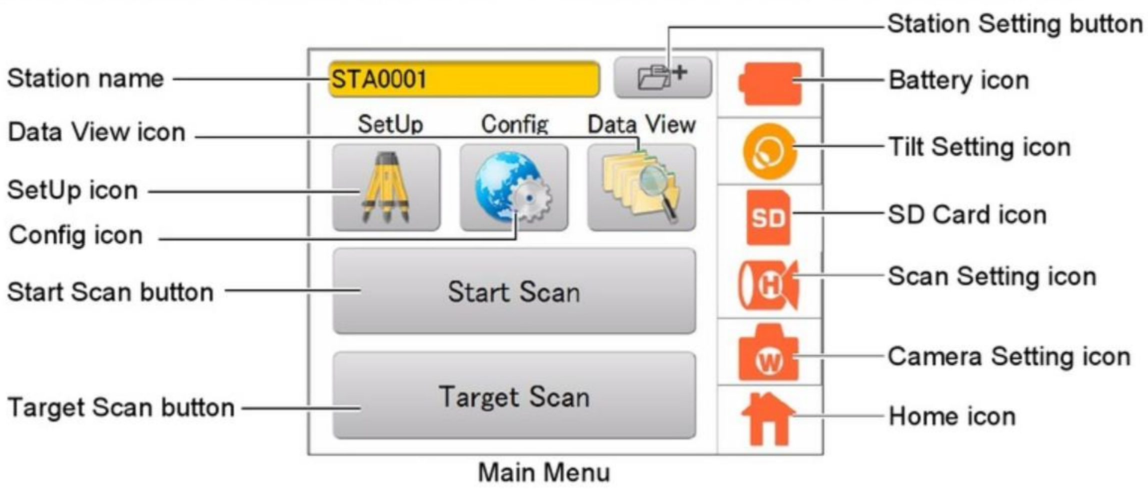

Figure 5.1: Settings for Topcon GLS 2000 [42]

An inspection on site is done before the scanning work. The bridge is located in city center with a high traffic from pedestrians on and vehicles under the bridge. The inspection resulted in making 14 scans.

The inspection and study of the geometry and location of the bridge has been significant to plan the scanning work in an effective way. The placement of the scanner is important to capture all necessary points that create the point cloud. This has been quite challenging due to high traffic from pedestrians and vehicles. Figure 5.2 shows the placement of scanner from different positions and angles.

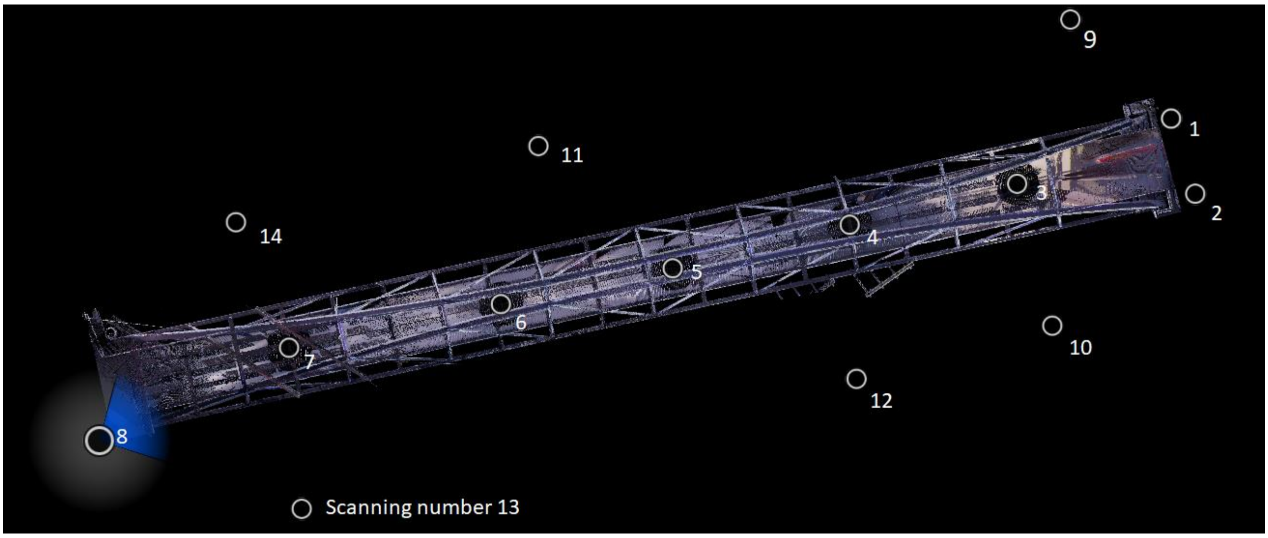

Figure 5.2: Placement of the scanner around and inside the bridge

The real picture of on-site scanning for the placement number 13 is shown in figure 5.3 as an example.

Figure 5.3: Real picture of on-site scanning

The target sheets and surrounding environment is used to merge the various scans into one unified point cloud. A total number of 20 target sheets is used for this purpose. The target sheets were spread throughout the bridge horizontally and vertically.

Next step is to merge the scans. Autodesk Recap is used to merge all 14 scans taken. After merging the scans, the unified 3D point cloud of the site is created. This process is called point cloud registration and merging. The point cloud of the bridge with surroundings is shown in figure 5.4.

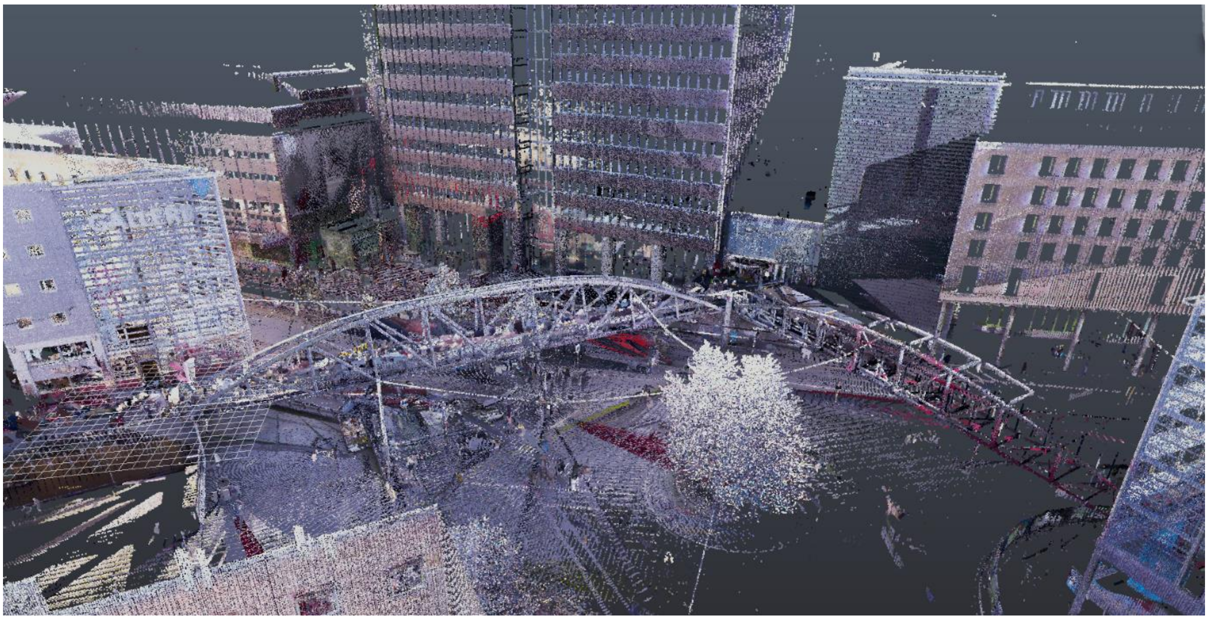

Figure 5.4: Unified point cloud of the site

The unified point cloud contains unnecessary points as shown in figure 5.4. Therefore, the next step was to eliminate and reduce unnecessary points to get an as accurate as possible 3D point cloud of the bridge. This process is called noise reduction. Autodesk recap has been used for this purpose. The process of noise reduction became heavy due to high pedestrian traffic on the bridge and the crowded surrounding area around the bridge itself. Figure 5.5 shows the point cloud after noise reduction.

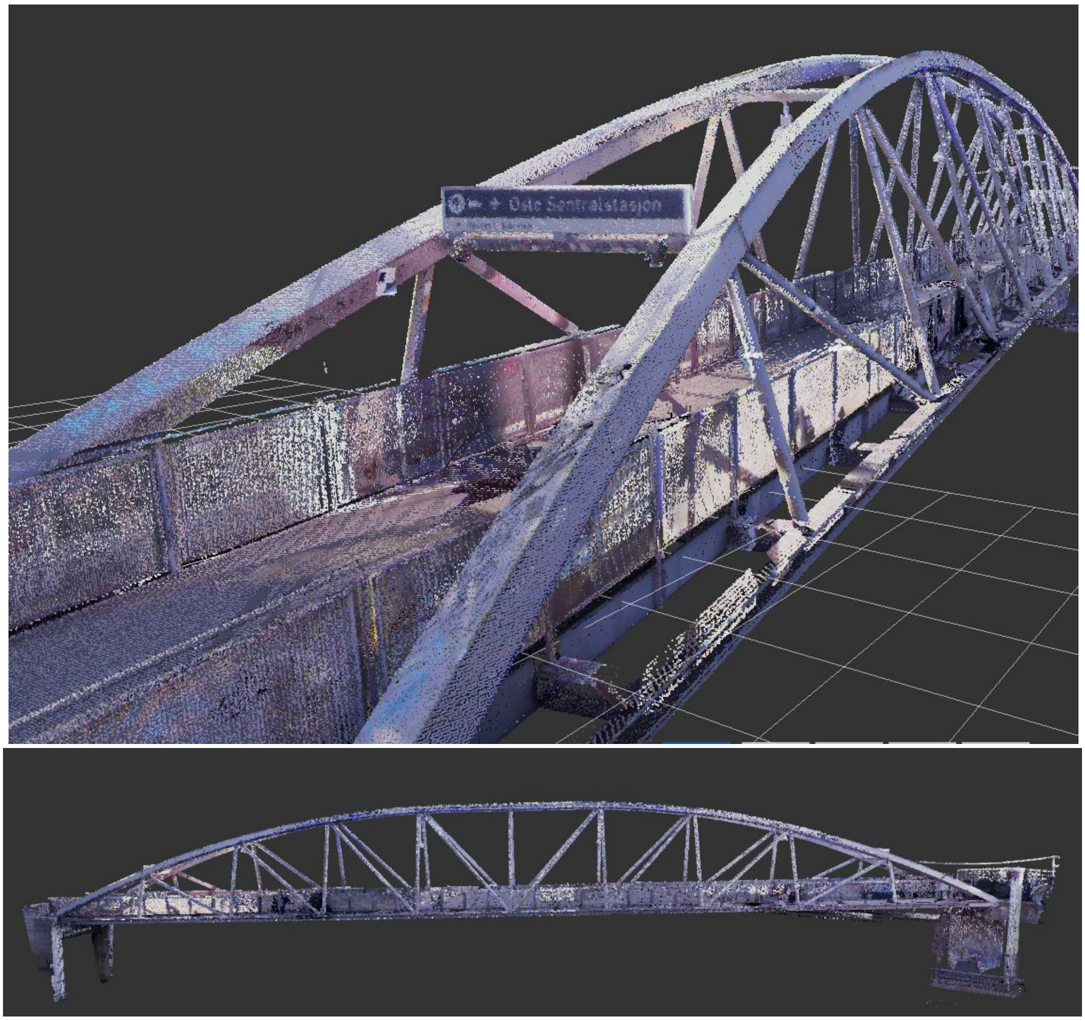

Figure 5.5: Point cloud of the pedestrian bridge after noise reduction

The next step in data processing of the point cloud is to check the accuracy of the point cloud. The accuracy of the point cloud can be checked in software tools such as Autodesk Recap, CloudCompare and Geomagic Control X. It is possible to inspect the denoised model from various angles to check the accuracy of the points that are from various components.

Next step in processing data from point cloud is to segment various components of the structure. The purpose of segmentation is to prepare components for the algorithms. The segmentation can be done in both Geomagic Control X and CloudCompare. The very first segmentation is to separate the left side and right side of the bridge as shown in figure 5.6.

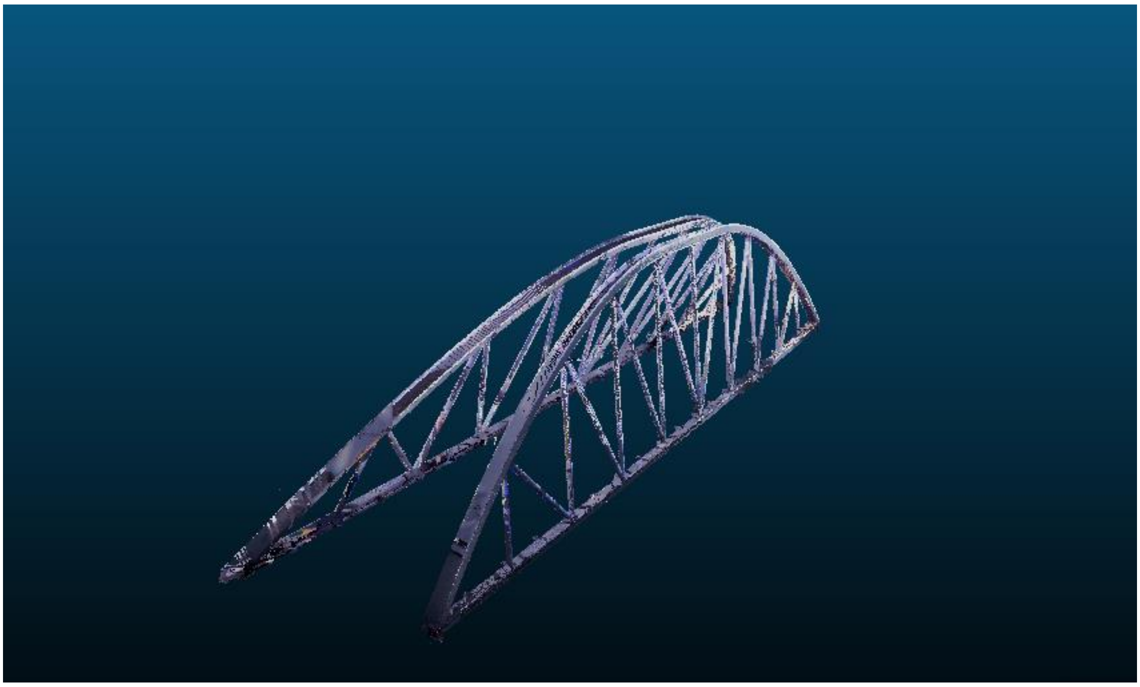

Figure 5.6: Segmentation of the bridge to separate left side and right side

As shown in figure 5.6 and 5.7, the left and right side of the bridge are bent inward. For the convenience of the algorithms, it is tried to rotate both side of the bridge in Geomagic Control X so that both sides are parallel to X-Z plane. For this purpose, the left side of the bridge is rotated by -15,5 degrees on X-axis and by -11,9 degrees on Z-axis. The right side of the bridge is rotated by 15,8 degrees on X-axis and by -12,7 degrees on Z-axis.

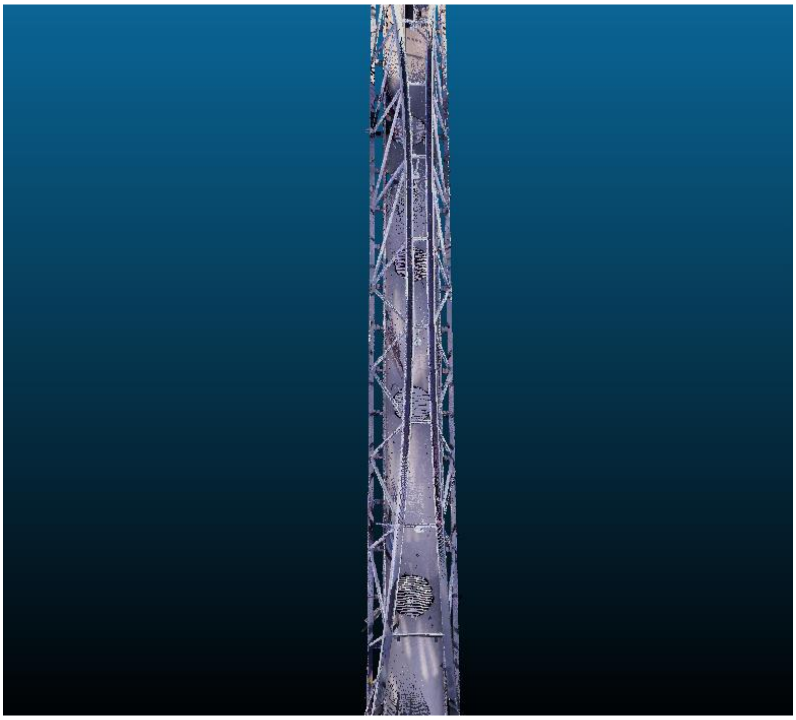

Figure 5.7: The point cloud shown from above to illustrate the curve inward

Picture 5.8 shows both sides of the bridge after the rotation. The sides are now parallel to the X-Z plane.

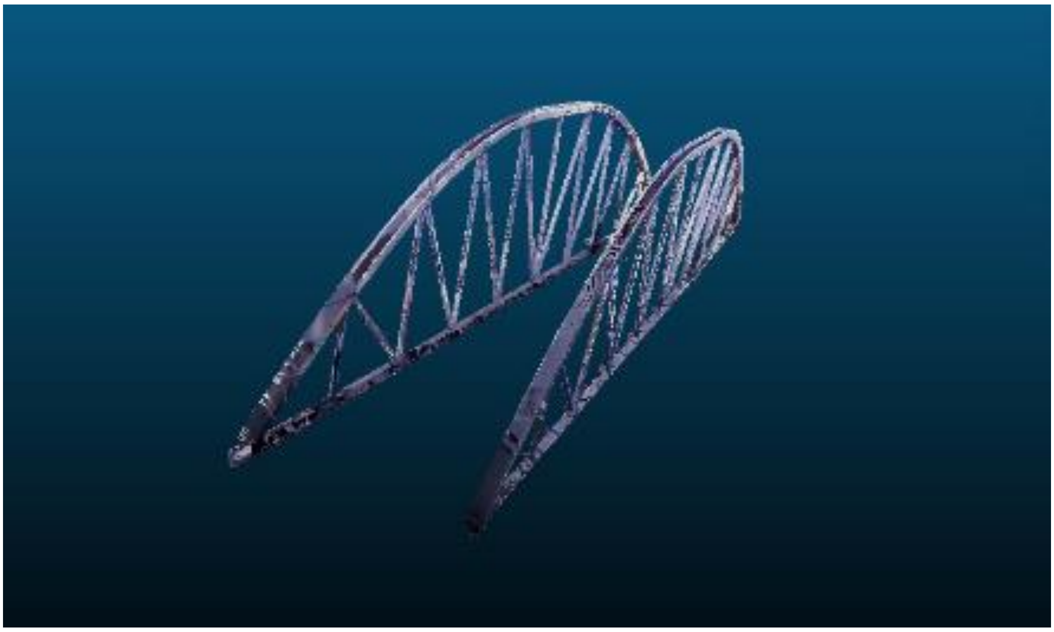

Figure 5.8: The rotate sides of the bridge parallel to X-X plane

The main components to be run through the algorithms are I-shaped steel member in left and right top chords and the circular pipe components on both sides. The process described here is applied for both sides separately.

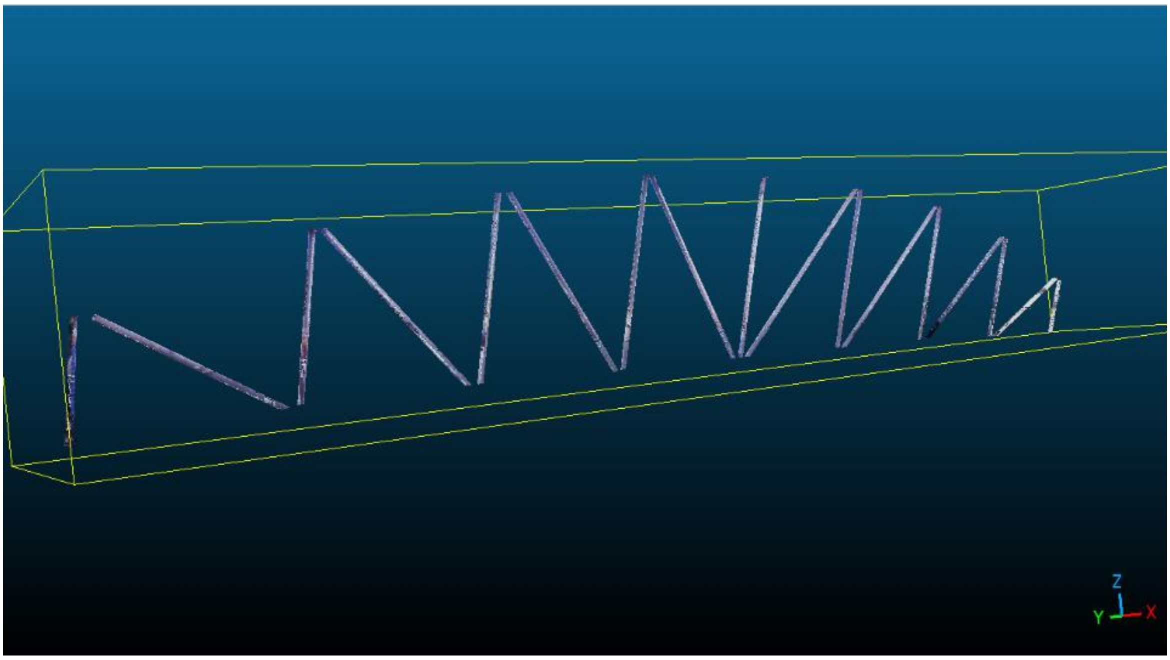

Figure 5.9: Picture of circular pipe components segmented in CloudCompare

An example of segmentation is figure 5.9 which shows circular pipe components on the left side of the bridge in point cloud that are segmented manually using CloudCompare. The top and bottom of the joints between two pipe components are removed so that each component can be run separately through algorithm. After the segmentation, the 3D data from the point cloud can be extracted to a PCD-file before importing to Matlab.

The segmentation and data extraction of top chords is done in Geomagic Control X. The cross section of I-shaped steel member in top chord is complicated. The I-shaped top chord member is also adjusted to be parallel to x-z plane.

After that, the web point cloud which is shown in red color can be selected manually. The data can then be extracted to a text-file.

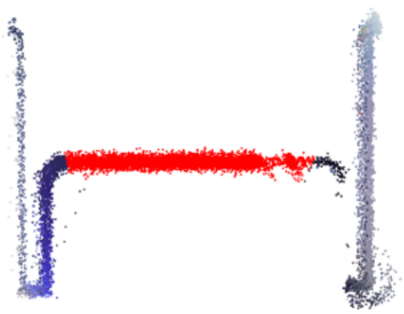

Figure 5.10: Selecting the web point cloud (red points)

The last step for data extraction is to find parameters of I-shaped steel members in top and bottom chord. The necessary data to use in families of the I-shaped members in parametric model in Dynamo are extracted from the point cloud in Geomagic Control X and are shown in figure 5.11 and 5.12. Since only one side of the I-beam can be scanned in top chord in figure 5.11, it is assumed that the thickness of web is equal to d1.

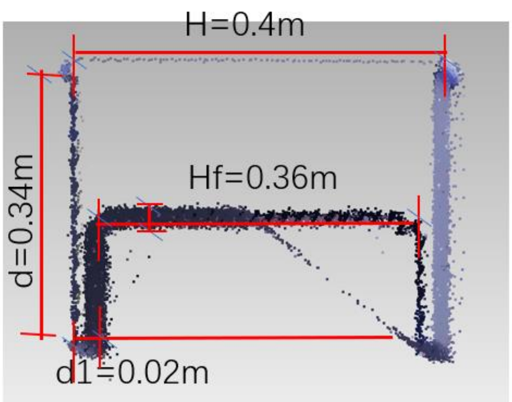

Figure 5.11: Data extraction of I-shaped steel member in top chord

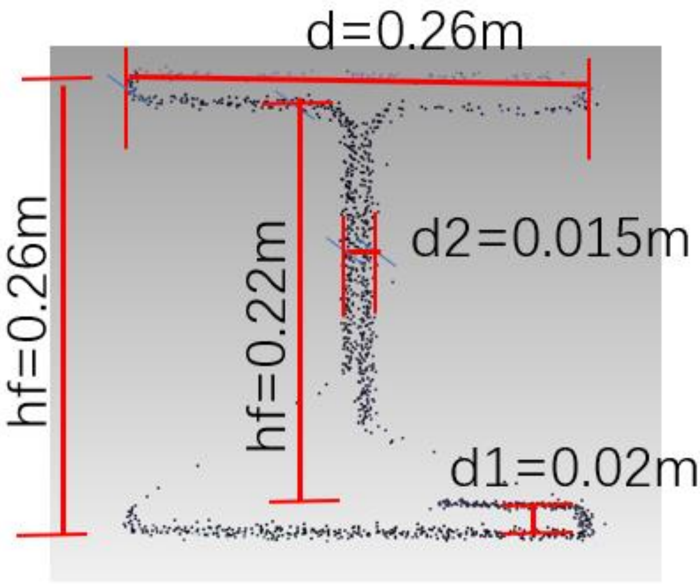

Figure 5.12: Data extraction of I-shaped steel member in bottom chord

5.2 MATLAB and Algorithms

The next step in the method used in this thesis is to extract the geometry information of components of the bridge. The point cloud available is a huge number of points in 3D which are captured and gathered from various parts of the bridge. These point clouds need to be run through some algorithms so that the algorithm recognizes the pattern of point cloud and extract the information about the geometry of a component point cloud. Various software programs such as Python or MATLAB can be used for this purpose. The program chosen in this thesis is MATLAB, and the algorithms made in this program are presented in this chapter. The main components of the pedestrian bridge that are run through algorithms are curved I-shaped steel member in top chords and circular pipe diagonal struts on both sides of the bridge.

5.2.1 Polynomial Curve Fitting Algorithm

The polynomial fitting algorithm is mainly used to extract geometry information of the top chords of pedestrian bridge. The complete MATLAB code developed for this purpose can be found in appendix part A. The cross-section shape of I-shaped steel member seems to be complex, and the top chord is curved in addition. The idea behind extraction of geometry information is to find the accurate curve shape of the I-shaped steel member and find the design section size closest to the cross-section of the I-shaped steel member.

The polynomial curve fitting algorithm is used in this case to find the degree of the best polynomial fitting of the web point cloud. The web point cloud data are extracted from Geomagic Control X and includes a text file that contains the x- y- and z-data of the points in the web point cloud. The process described here applies the top chords on both sides of the bridge separately.

To find the best curve fitting for the shape of the top chord, z coordinates and x coordinates of top chord are extracted from the point cloud. Some function in MATLAB can return the coefficients for a polynomial p(x) of degree n that is a best fit for the data in z.

The polynomial p(x) is defined as below:

$$p(x) = p1 \cdot x^n + p2 \cdot x^{n-1} + \cdots + p_n \cdot x + p_{n+1}$$

After testing various degrees of polynomials, it seems that the fitting effect of 10^{th} degree polynomial is relatively good.

The polynomial p at each point in x is then evaluated to get the fitted value. Figure 5.13 shows the values for x- and z-coordinates of the web point cloud. The fitted values after defining the polynomial of 10^{th} degree are shown in figure 5.14. Figure 5.15 shows how accurate the fitted values are to the values obtained from web point cloud.

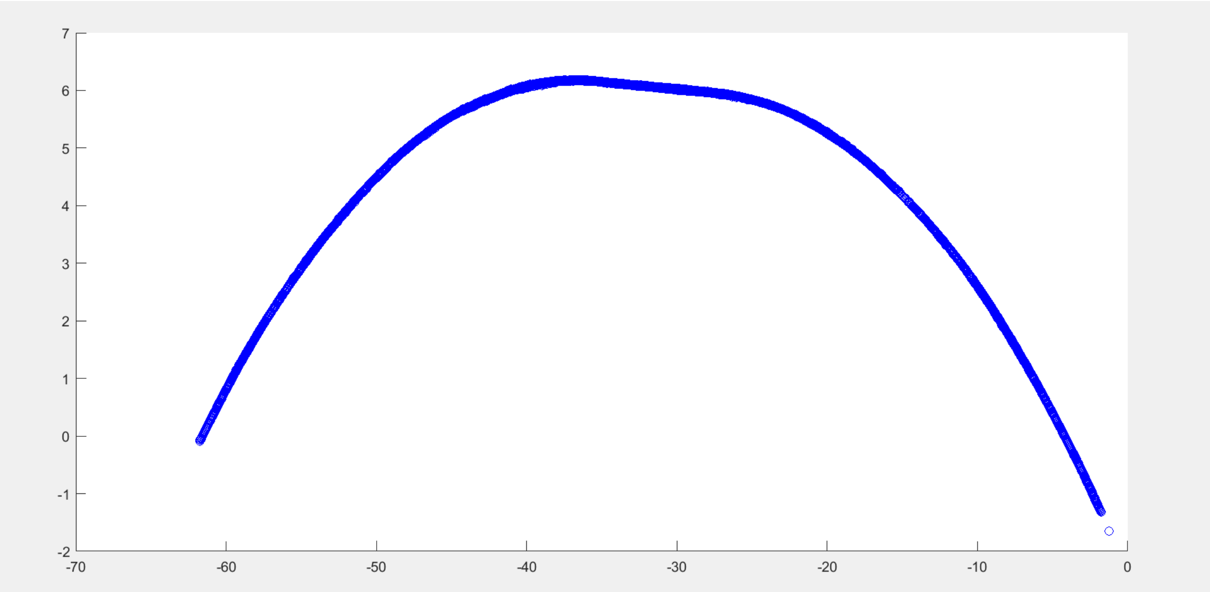

Figure 5.13: The web point of the top chord

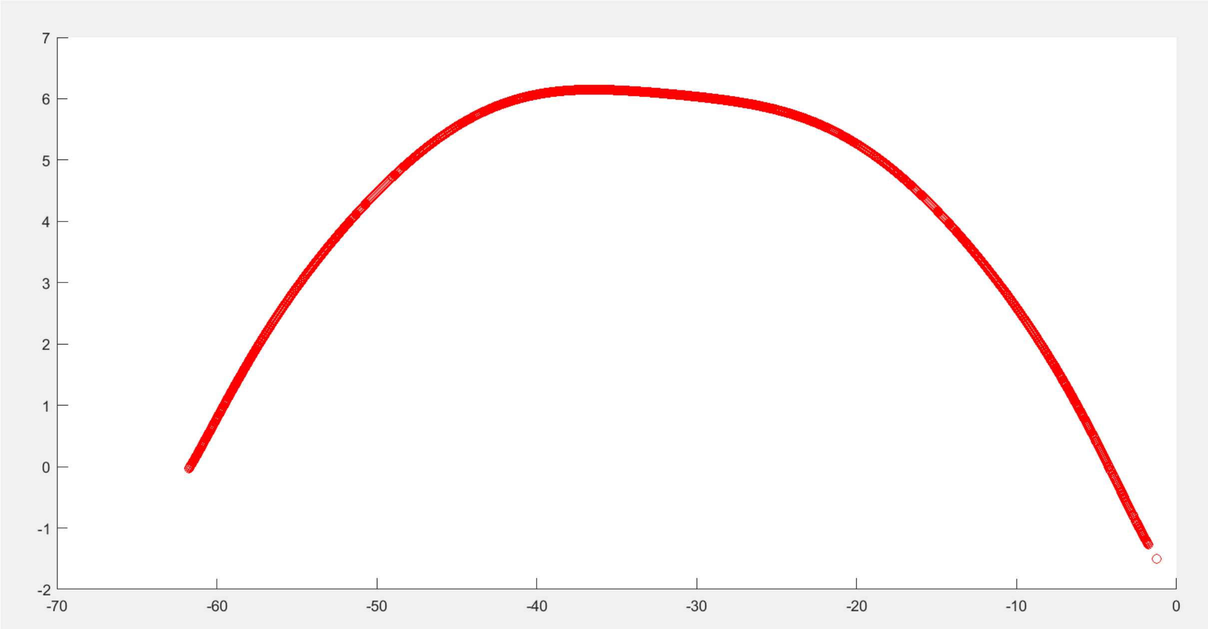

Figure 5.14: Fitted values based on the polynomial

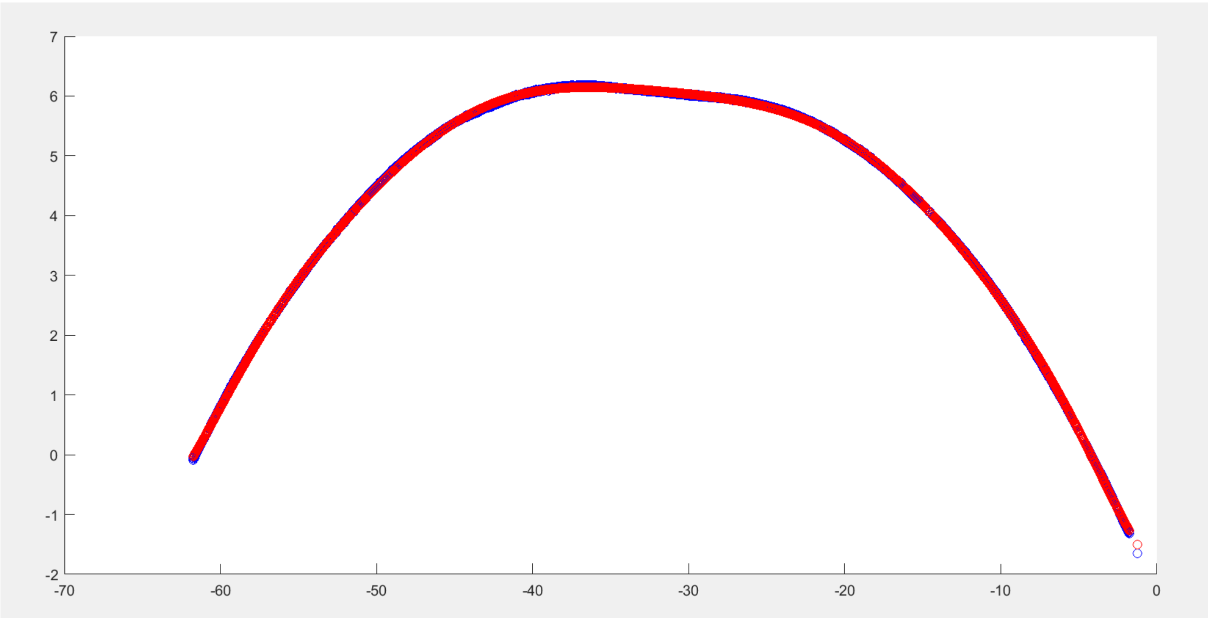

Figure 5.15: Fitted values and the web point of the top chord

The sliding window algorithm is used in the next step to deal with the data. The interval length of d=2 is chosen for the sliding window algorithm. The value of the insertion point of the sliding algorithm is then defined based on d and minimum and maximum values of x-coordinates.

By derivation of the fitting polynomial, the insertion point derivative of the sliding widow algorithm can be achieved. The z value of the fitting polynomial at insertion points can then be calculated by evaluating the polynomial p at each point in x insertion points. The slope of insertion points of sliding window algorithm can then be decided using the derivation of the polynomial at x insertion points. This is because the derivative of a function shows the slope of a tangent line to the curve at any instant. By using the inverse tangent of the slope of insertion points of sliding window, the radians of the fitting polynomial can be obtained.

A loop is created to deal with the sliding window algorithm.

The web point is then rotated on xz-plane. The purpose is to rotate the web so that the center of web points can be detected. For this purpose, a rotation matrix consisting of the angle of rotation at the i^{th} interpolation point is used.

$$R_{xz} = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix}$$

The rotation matrix helps to rotate with the i^{th} interpolation point as the origin to make the segment point cloud parallel to the x-axis.

Figure 5.16 shows the results of sliding window algorithm with points indicating center of mass.

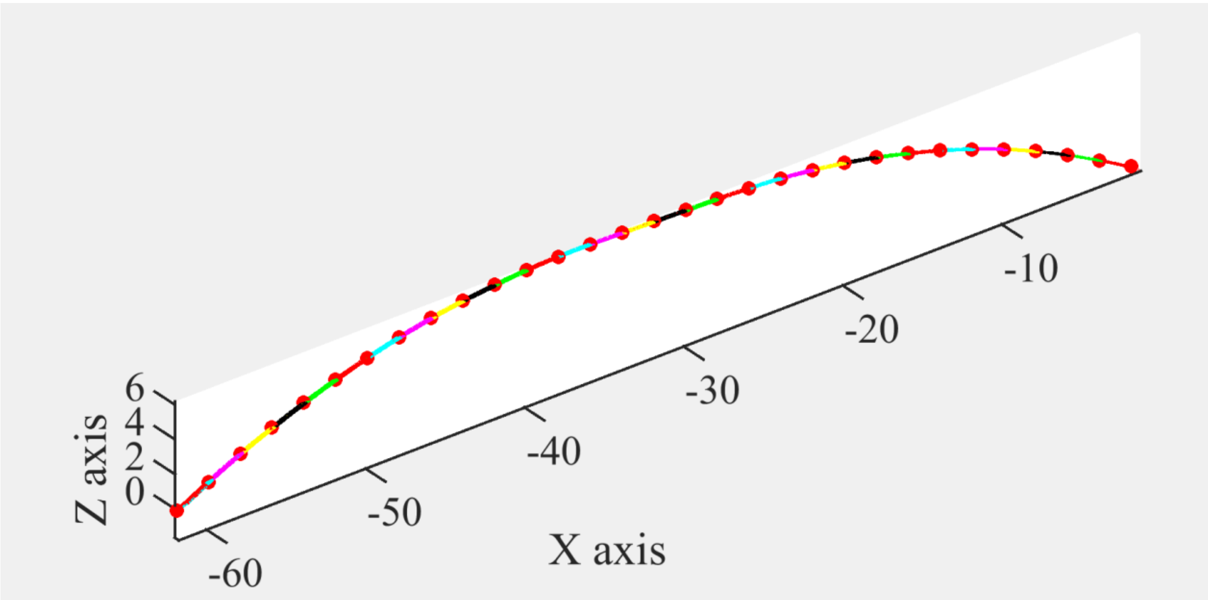

Figure 5.16: Results after sliding window algorithm

Figure 5.17 shows the X- and Z insertion points. The coordinates of insertion points can be extracted for further work with parameters to deal with the top chords.

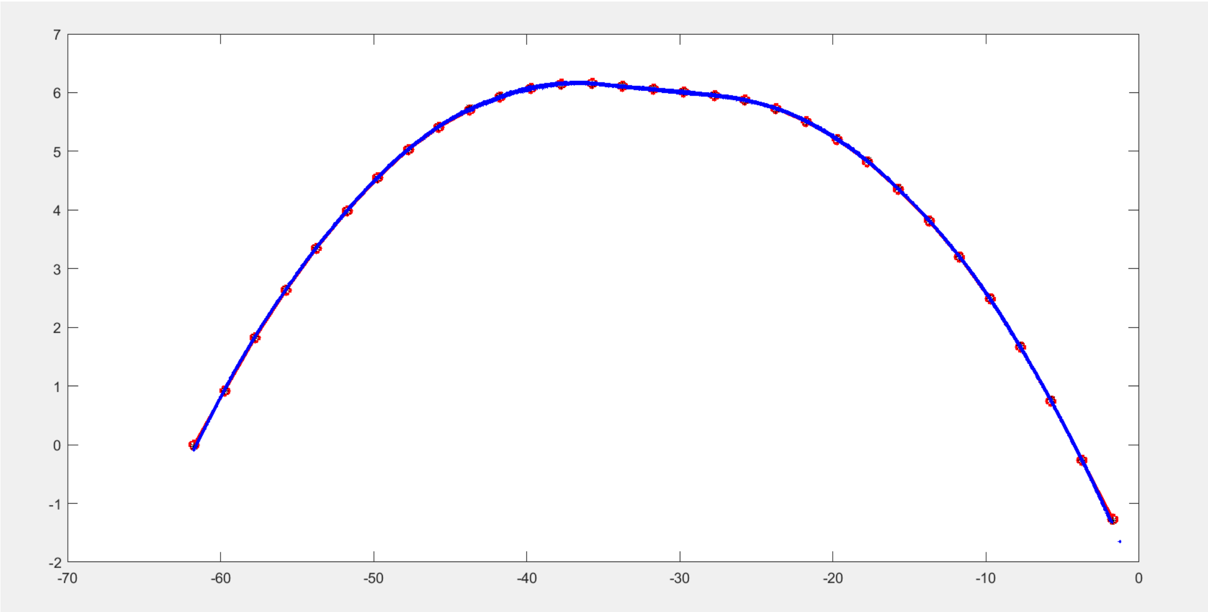

Figure 5.17: Insertion points

The coordinates for the insertion points of the left top chord and right top chord are shown in table 1 and table 2 respectively.

x	y	z
-61734	5579.147	-24.6849
-59734	5579.147	917.5323
-57734	5579.147	1839.03
-55734	5579.147	2653.791
-53734	5579.147	3363.354
-51734	5579.147	3988.876
-49734	5579.147	4543.152
-47734	5579.147	5025.3
-45734	5579.147	5426.085
-43734	5579.147	5736.118
-41734	5579.147	5952.463
-39734	5579.147	6081.685
-37734	5579.147	6139.086
-35734	5579.147	6145.013
-33734	5579.147	6119.659
-31734	5579.147	6077.949
-29734	5579.147	6025.841
-27734	5579.147	5958.958
-25734	5579.147	5863.855
-23734	5579.147	5721.584
-21734	5579.147	5512.628
-19734	5579.147	5221.855
-17734	5579.147	4841.94
-15734	5579.147	4373.892
-13734	5579.147	3823.913
-11734	5579.147	3197.008
-9733.97	5579.147	2489.551
-7733.97	5579.147	1685.573
-5733.97	5579.147	764.9469
-3733.97	5579.147	-264.032
-1733.97	5579.147	-1289.77

Table 1: Insertion points of left top chord

x	y	z
-61665.1	683.055	1713.75
-59665.1	683.055	2625.494
-57665.1	683.055	3556.91
-55665.1	683.055	4384.772
-53665.1	683.055	5098.765
-51665.1	683.055	5720.694
-49665.1	683.055	6268.354
-47665.1	683.055	6745.642
-45665.1	683.055	7145.876
-43665.1	683.055	7459.693
-41665.1	683.055	7682.34
-39665.1	683.055	7817.807
-37665.1	683.055	7879.148
-35665.1	683.055	7885.574
-33665.1	683.055	7857.605
-31665.1	683.055	7811.823
-29665.1	683.055	7756.642
-27665.1	683.055	7690.157
-25665.1	683.055	7600.565
-23665.1	683.055	7469.049
-21665.1	683.055	7274.408
-19665.1	683.055	6998.236
-17665.1	683.055	6629.164
-15665.1	683.055	6164.724
-13665.1	683.055	5609.798
-11665.1	683.055	4971.553
-9665.13	683.055	4252.246
-7665.13	683.055	3443.488
-5665.13	683.055	2528.507
-3665.13	683.055	1502.783
-1665.13	683.055	428.2373

Table 2: Insertion point of right top chord

5.2.2 Euclidean Clustering Algorithm

The data to be extracted from the point cloud for circular pipe components are for example intersection points, center point of each end of a circular pipe component, and the radius of each circular pipe component. Obtaining these points and data from a point cloud seems to be time consuming, since every component needs to be segmented manually in point cloud processing software tools such as Geomagic Control X or CloudCompare.

With a bridge structure made of 17 circular pipe components on each side, the process of manual segmentation and input of data for each component will be heavy and time consuming. Hence, the research in this thesis includes identifying an algorithm that recognizes components by clustering. The algorithm used and developed in this case is Euclidean Clustering algorithm which is an unsupervised machine learning algorithm. The algorithm and process described here applies to the circular pipe components on both sides of the bridge separately.

The algorithm is used to extract geometry information of circular pipe elements of the bridge. The algorithm segments the point cloud into clusters with a specified minimum Euclidean distance between points from different clusters. The algorithm provides each point in the point cloud an integer cluster label and returns the label of all points and the number of clusters. The data obtained from data processing phase has been used for this algorithm. After the manual segmentation of each side of the bridge, the circular pipe diagonal struts are extracted out of the point cloud. So that the algorithm recognizes and clusters the components more easily, the joints of the diagonals are cut, and the neighbor diagonals are separated.

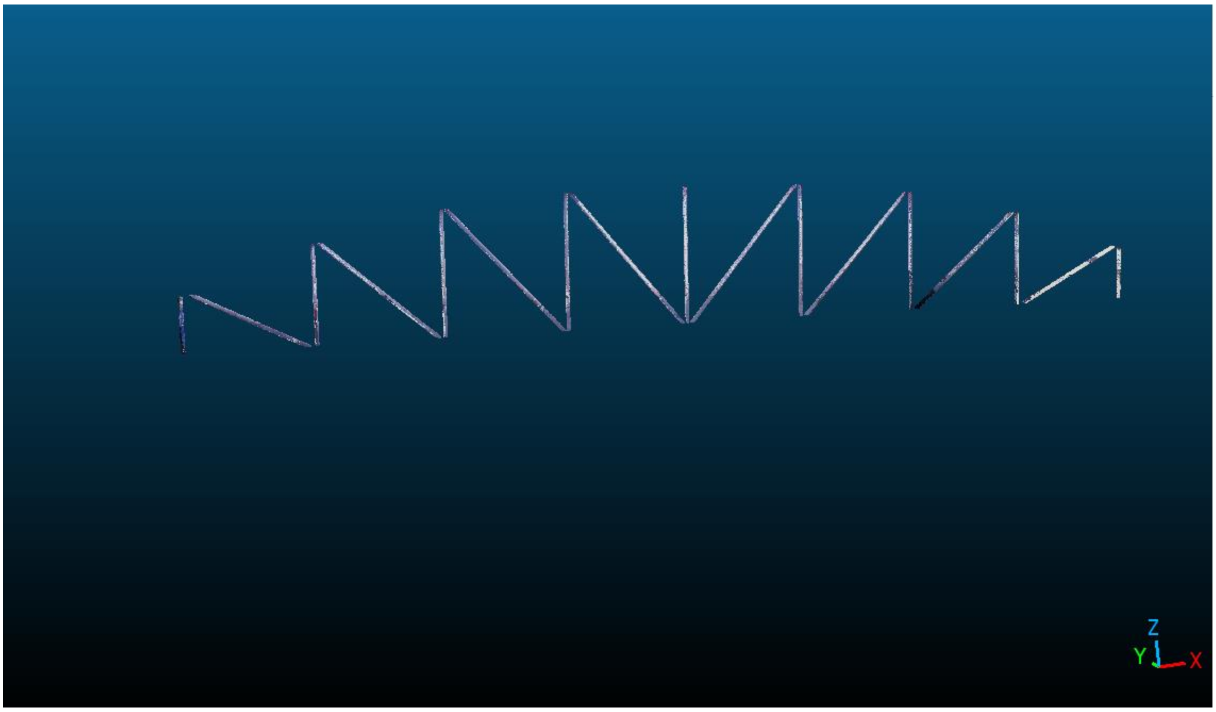

Figure 5.18: Segmented point cloud to be used in the Euclidean Clustering algorithm

The PCD-file, which is a file format for storing 3D point cloud data, obtained from these circular pipe components is then imported into MATLAB for further work through the algorithm. The imported point cloud is shown in figure 5.19.

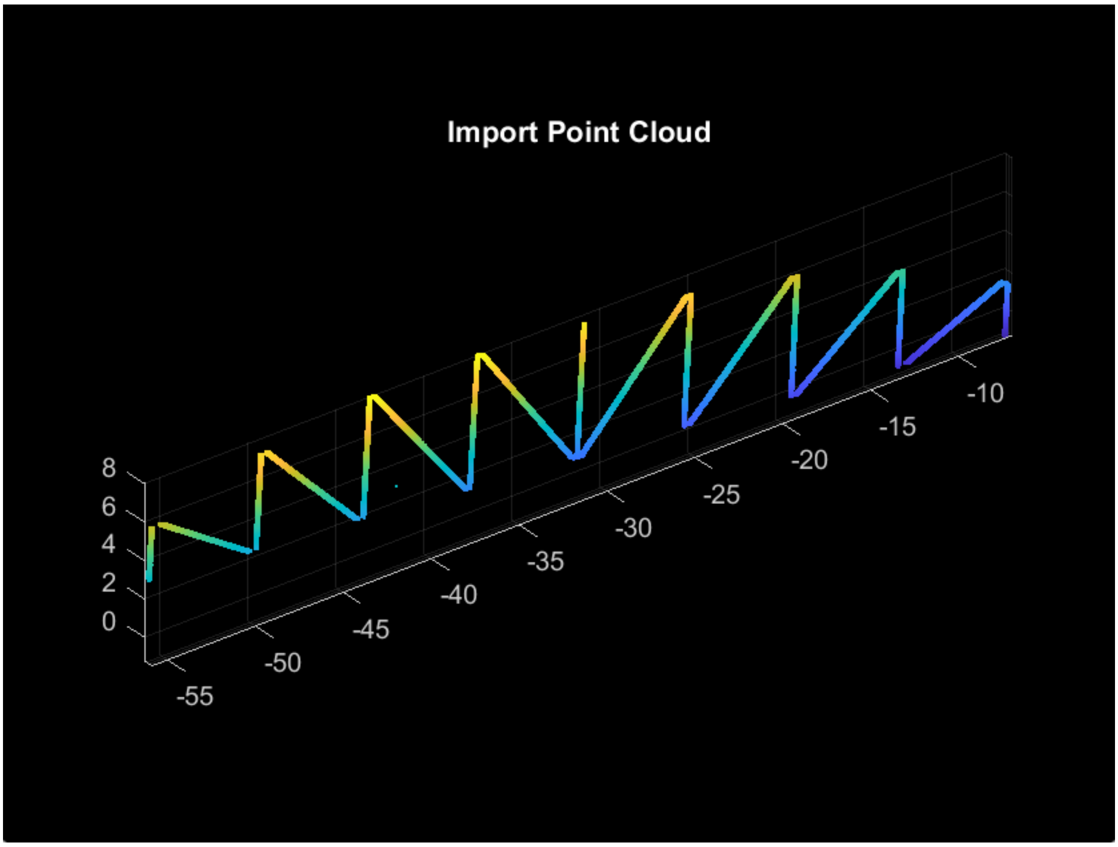

Figure 5.19: Imported point cloud of circular pipe components from PCD-file

Since the algorithm segments the point cloud into clusters, a threshold needs to be defined. This threshold is set to be 0,06 as a minimum Euclidean distance between two distinct cluster points. Figure 5.20 shows the result of segmentation. The principle of distance correlation to cluster the point cloud is used in this case. In other words, if the distance between point clouds exceeds a certain threshold, it can be read as they are not the same point cloud

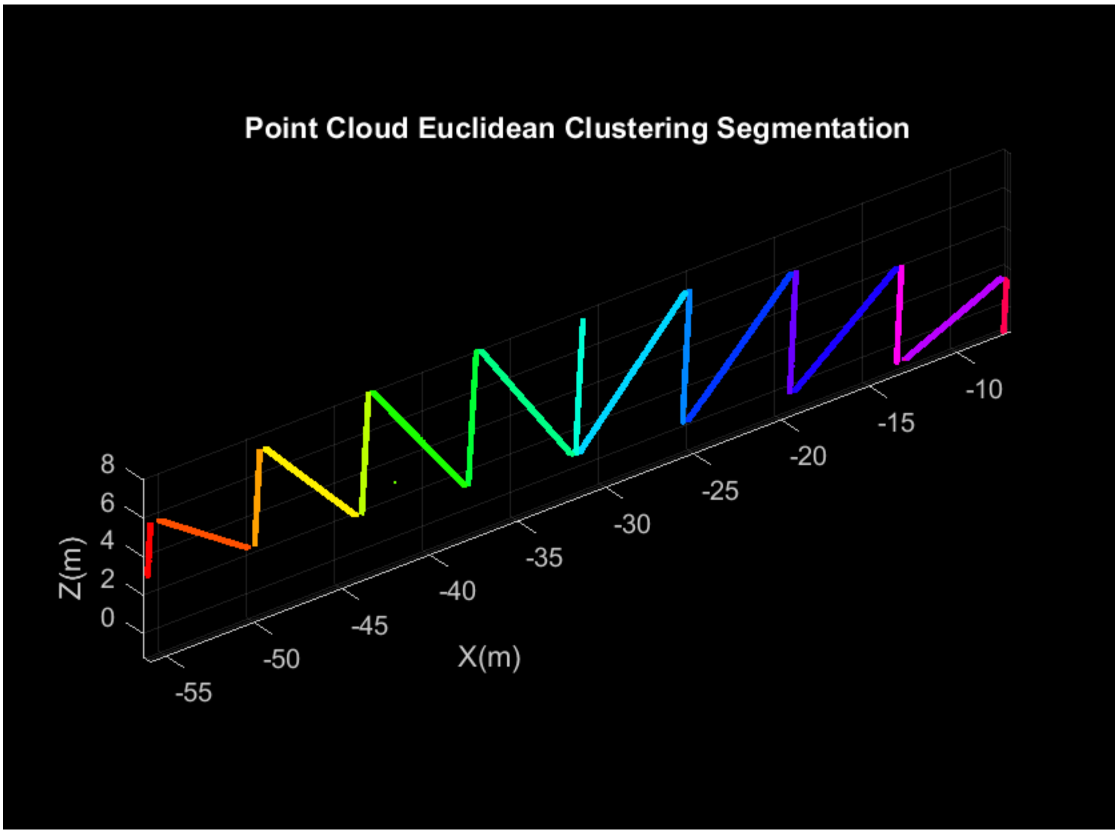

Figure 5.20: Result of segmentation in Euclidean clustering algorithm

In order to reduce the influence of noise on the fitted cylinders, a maximum distance from the point to the cylinder surface is defined and is set to be 0,005. The segmentation result is then adjusted to appear in all three axes' as shown in figure 5.21.

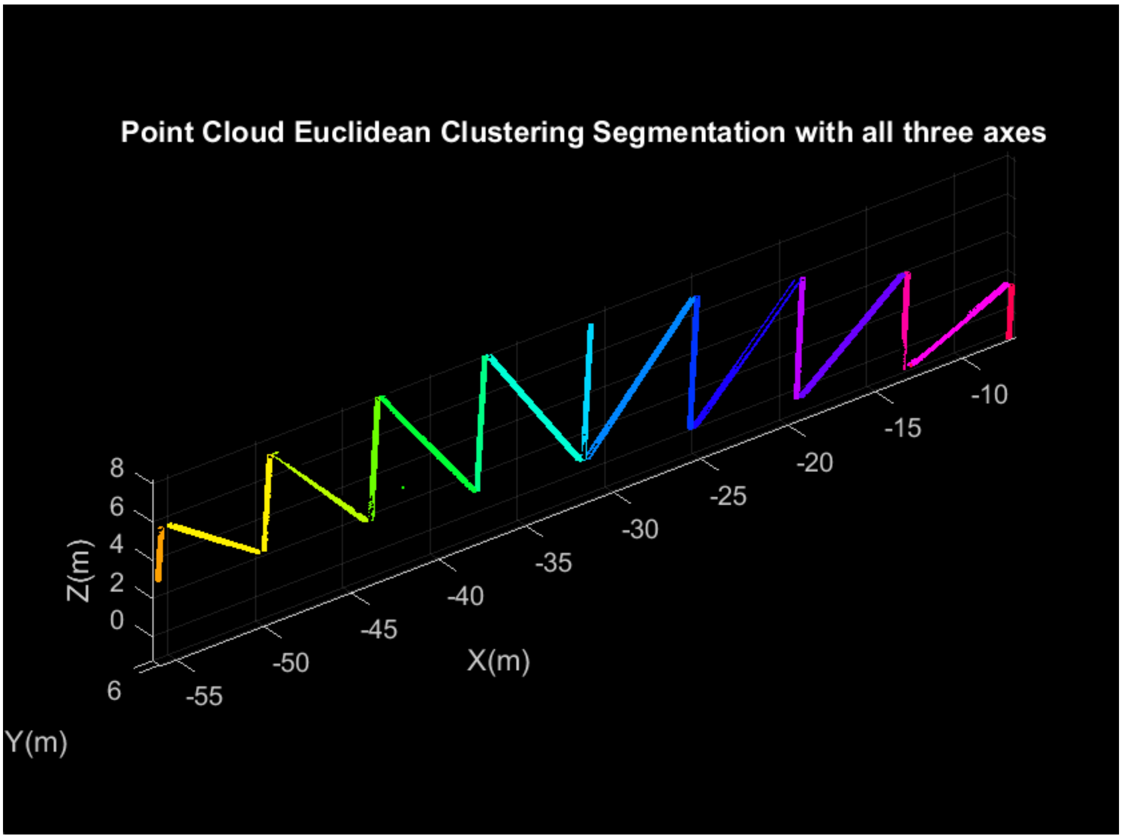

Figure 5.21: Result of segmentation in three axis

The final segmentation result with centerlines of each component which finds the intersection points is shown in figure 5.22. The centerlines fitted to cylinders are shown in red.

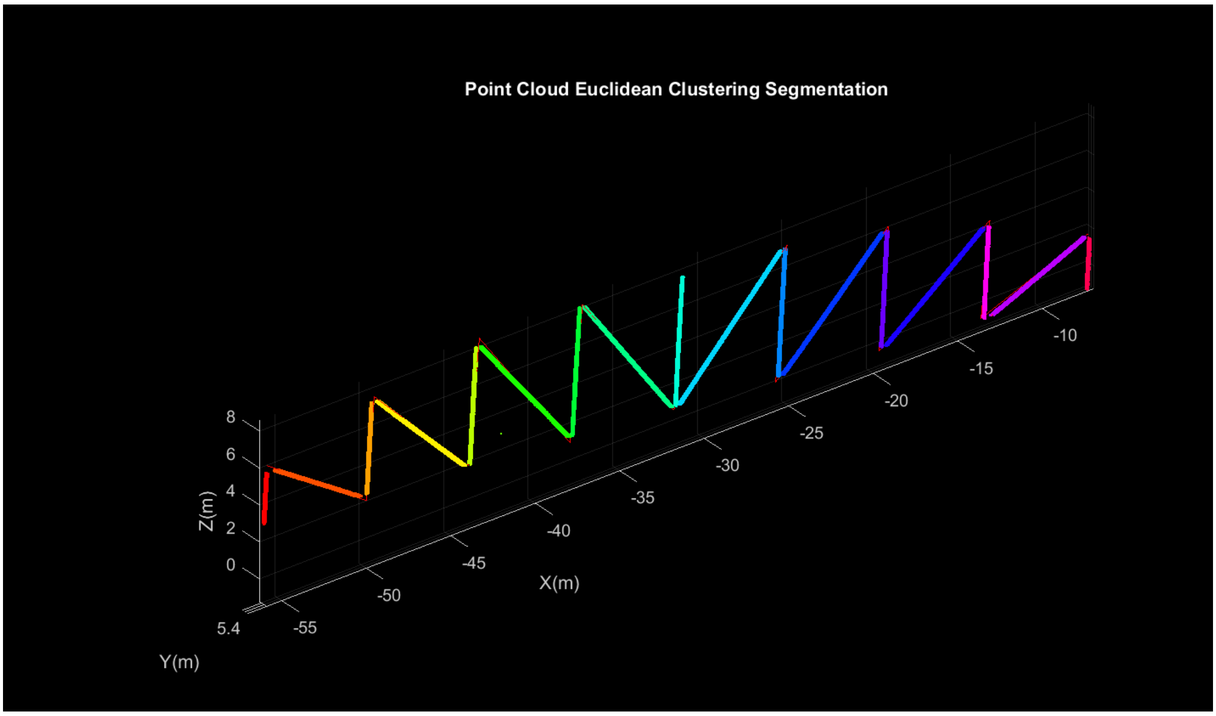

Figure 5.22: Result of segmentation with centerlines and intersections

The complete MATLAB code developed for this purpose can be found in appendix part B. The parameters to be extracted through the algorithm are center point of each circular end of a cylinder, center point of each cylinder, orientation, radius, and length of each cylinder. The cylinder model parameters in MATLAB sorted as a 1-by-7 scalar vector that describes a cylinder has been utilized. The vector is in form of [x1, y1, z1, x2, y2, z2, r], where [x1, y1, z1] and [x2, y2, z2] are the centers of each end-cap surface of the cylinder, and r is the radius of the cylinder. The results in form of cylinder model parameters obtained from the algorithm is shown in table 3. The order of cylinders is shown in figure 5.23. Number of cylinders in result tables corresponds to the order of cylinders.

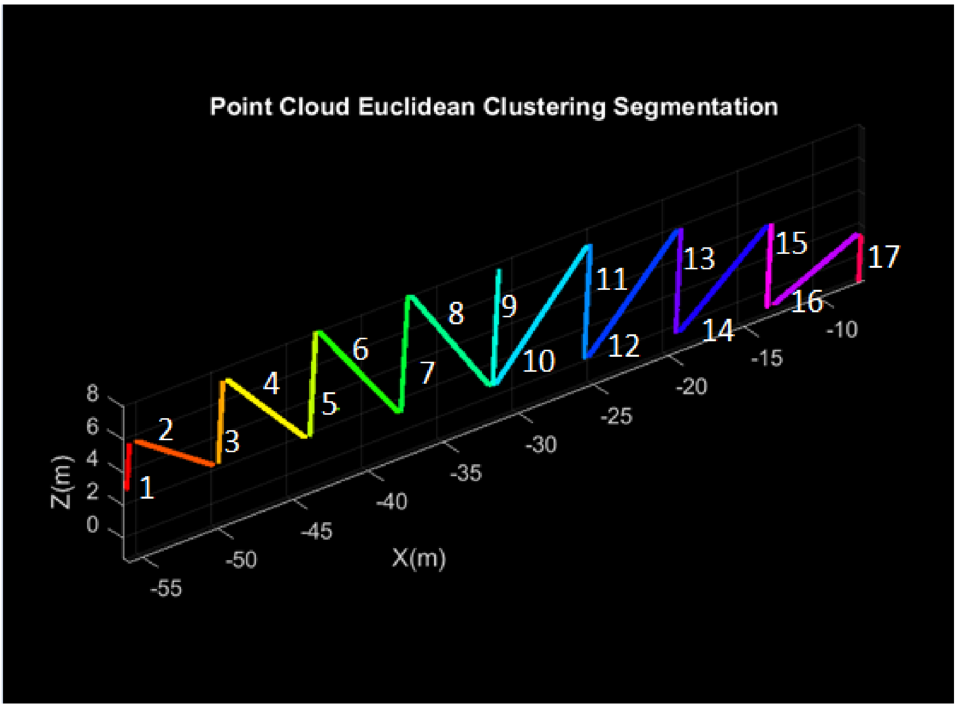

Figure 5.23: Order of cylinders

Cylinder number	**Cylinder model parameters** **[x1 y1 z1 x2 y2 z2 r]**
1	[-55.6265 5.8076 5.6596 -55.7853 5.7480 3.1247 0.0979]
2	[-54.9580 5.8121 5.6925 -50.2789 5.6785 2.5357 0.0717]
3	[-49.6928 5.6767 2.5929 -49.3141 5.8308 7.2552 0.0948]
4	[-48.9756 5.8011 7.3315 -43.9515 5.7852 2.1995 0.1519]
5	[-43.6993 5.6701 2.0523 -43.1066 5.8572 8.0464 0.0806]
6	[-37.9680 5.7566 1.4677 -42.6963 5.8884 8.1150 0.1533]
7	[-37.6408 5.6476 1.4835 -36.9712 5.8564 8.0570 0.0669]
8	[-31.7676 5.6435 0.9717 -36.7964 5.8288 8.0807 0.0593]
9	[-31.5431 5.5880 0.9699 -30.9264 5.9553 7.6065 0.0673]
10	[-31.3090 5.6176 0.9424 -25.1196 5.8386 6.9842 0.0684]
11	[-25.4814 5.6782 0.4744 -24.8660 5.8175 6.9898 0.0700]
12	[-25.1665 5.6138 0.4018 -19.1133 5.8047 5.9280 0.0805]
13	[-19.0714 5.5529 -0.1675 -13.3155 5.7484 4.0368 0.0649]
14	[-19.3708 5.5789 -0.1123 -18.8782 5.7894 5.8542 0.0719]
15	[-12.5361 5.6796 -0.4422 -7.2704 5.6444 1.5114 0.0888]
16	[-13.3588 5.5889 -0.6864 -12.8732 5.7600 4.0471 0.0922]
17	[-7.1867 5.5669 -1.2397 -7.0159 5.6454 1.4044 0.0711]

Table 3: Cylinder model parameters for the 17 cylinders on left side

Table 4 shows the other parameters of each cylinder.

Cylinder number	Center of cylinder [x y z]	Orientation	Length(m)
1	[-55.7059 5.7778 4.3921]	[-0.1589 -0.0596 -2.5349]	2.5405
2	[-52.6184 5.7453 4.1141]	[4.6791 -0.1336 -3.1568]	5.6460
3	[-49.5034 5.7538 4.9240]	[0.3787 0.1540 4.6623]	4.6801
4	[-46.4636 5.7931 4.7655]	[5.0241 -0.0159 -5.1319]	7.1818
5	[-43.4029 5.7636 5.0493]	[0.5928 0.1871 5.9941]	6.0263
6	[-40.3321 5.8225 4.7914]	[-4.7283 0.1318 6.6473]	8.1585
7	[-37.3060 5.7520 4.7702]	[0.6695 0.2088 6.5735]	6.6108
8	[-34.2820 5.7362 4.5262]	[-5.0288 0.1853 7.1090]	8.7098
9	[-31.2348 5.7717 4.2882]	[0.6167 0.3673 6.6366]	6.6753
10	[-28.2143 5.7281 3.9633]	[6.1894 0.2211 6.0418]	8.6522
11	[-25.1737 5.7479 3.7321]	[0.6155 0.1393 6.5154]	6.5459
12	[-22.1399 5.7093 3.1649]	[6.0532 0.1909 5.5261]	8.1985
13	[-16.1935 5.6507 1.9346]	[5.7559 0.1955 4.2042]	7.1305
14	[-19.1245 5.6841 2.8710]	[0.4926 0.2105 5.9665]	5.9905
15	[-9.9032 5.6620 0.5346]	[5.2656 -0.0352 1.9536]	5.6164
16	[-13.1160 5.6745 1.6803]	[0.4856 0.1711 4.7335]	4.7614
17	[-7.1013 5.6061 0.0824]	[0.1708 0.0785 2.6441]	2.6508

Table 4: Data for struts on left side of the bridge

The mean value of all radius values can be calculated to be 0,0705 m. The median is 0,075 m.

The segmentation algorithm results in 16 intersection points at each side of the bridge that are shown in figure 5.24.

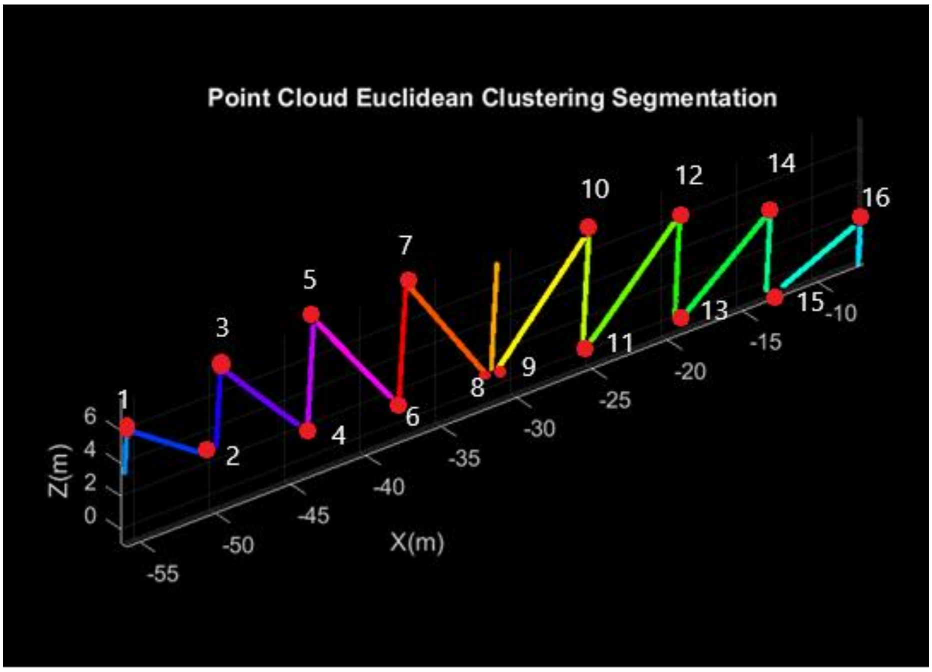

Figure 5.24: The position of intersection points between circular pipe diagonals

Cylinder model parameters for cylinder struts on right side are shown in table 5.

Cylinder number	Cylinder model parameters (Right side) [x1 y1 z1 x2 y2 z2 r]
1	[-55.7613 0.0875 2.8120 -55.5416 -0.0983 5.3859 0.0794]
2	[-55.1346 -0.0882 5.4735 -50.1863 0.1934 2.3916 0.0381]
3	[-49.6412 0.1318 2.3808 -49.2887 -0.1219 6.9484 0.0684]
4	[-49.0206 -0.1570 7.0582 -43.8810 0.2665 1.8874 0.0714]
5	[-43.5802 0.2124 1.9075 -43.1203 -0.1278 7.8337 0.0731]
6	[-37.7532 0.2797 1.5368 -42.8530 -0.1747 7.8325 0.0931]
7	[-37.5211 0.2814 1.4475 -36.9195 -0.0633 7.9359 0.0491]
8	[-31.6741 0.3518 1.1847 -36.7584 -0.1865 7.9829 0.1087]
9	[-31.1109 0.3068 1.1532 -26.3518 0.0270 5.5773 0.0928]
10	[-31.3401 0.3011 1.1018 -30.9723 -0.0972 7.5243 0.0833]
11	[-25.3193 0.3909 0.5784 -24.8302 0.0185 7.0417 0.0706]
12	[-24.9958 0.3966 0.5510 -19.0216 0.1793 6.0750 0.0605]
13	[-19.2270 0.4488 0.0262 -18.8187 0.1611 5.6962 0.0959]
14	[-19.0339 0.5052 -0.0240 -13.0730 0.2689 4.3940 0.0519]
15	[-13.1661 0.5959 -0.4402 -12.8304 0.2786 4.2968 0.0808]
16	[-12.8203 0.5312 -0.5031 -7.2026 0.4957 1.8832 0.0723]
17	[-7.1341 0.6647 -0.9322 -6.8987 0.4950 1.7098 0.0660]

Table 5: Cylinder model parameters for struts on right side

Table 6 shows other parameters obtained from MATLAB.

Cylinder number	Center of cylinder [x y z]	Orientation	Length(m)
1	[-55.6514 -0.0054 4.0989]	[0.2197 -0.1858 2.5739]	2.5900
2	[-52.6605 0.0526 3.9326]	[4.9483 0.2817 -3.0819]	5.8363
3	[-49.4649 0.0050 4.6646]	[0.3524 -0.2537 4.5676]	4.5882
4	[-46.4508 0.0547 4.4728]	[5.1396 0.4235 -5.1707]	7.3028
5	[-43.3502 0.0423 4.8706]	[0.4599 -0.3402 5.9262]	5.9537
6	[-40.3031 0.0525 4.6847]	[-5.0998 -0.4544 6.2957]	8.1148
7	[-37.2203 0.1091 4.6917]	[0.6017 -0.3446 6.4883]	6.5253
8	[-34.2163 0.0826 4.5838]	[-5.0843 -0.5383 6.7982]	8.5062
9	[-28.7313 0.1669 3.3652]	[4.7591 -0.2798 4.4241]	6.5038
10	[-31.1562 0.1019 4.3131]	[0.3678 -0.3983 6.4225]	6.4453
11	[-25.0747 0.2047 3.8101]	[0.4891 -0.3724 6.4632]	6.4924
12	[-22.0087 0.2880 3.3130]	[5.9742 -0.2174 5.5240]	8.1396
13	[-19.0229 0.3050 2.8612]	[0.4083 -0.2877 5.6700]	5.6919
14	[-16.0535 0.3870 2.1850]	[5.9610 -0.2363 4.4181]	7.4235
15	[-12.9982 0.4372 1.9283]	[0.3357 -0.3173 4.7370]	4.7594
16	[-10.0115 0.5134 0.6900]	[5.6177 -0.0355 2.3863]	6.1036
17	[-7.0164 0.5798 0.3888]	[0.2354 -0.1697 2.6420]	2.6579

Table 6: Parameters obtained from MATLAB for cylinder struts on right side

5.3 Dynamo and Parametric Driven BIM-model

The goal for this thesis is to find a way to measure the overall alignment of long-span spatial curved structures with circular pipe components quickly and accurately. After scanning the pedestrian bridge, some parameters are obtained from MATLAB based on the point cloud. These parameters will be used to generate a parametric model of the existing bridge. There are many software tools available for this purpose. The software tool used in this thesis is Dynamo. There are various ways to design a model based on parameters in Dynamo. This subchapter describes the parameters that are extracted from MATLAB and used as input to Dynamo, as well as the techniques of creation of the BIM-model of the pedestrian bridge.

5.3.1 Parameters

Various parameters are obtained through the geometry information extraction of the point cloud in MATLAB. The bridge components that are in focus under this research are curved I-shaped top chord and circular pipe struts.

For the circular pipe struts, the intersection points are used as the basis. There are 16 intersection points with x- y- and z-coordinates as shown in figure 5.25. However, there are 3 struts that are in a special position. These struts are marked in figure 5.25. Since there should be coordinates of two separate points at each end of a strut for the design, the coordinates of center point for the bottom end of the strut A and C can be calculated. Since these struts are positioned vertically, the x- and y-coordinates of the bottom of the struts can be assumed to be the same as the x- and y-coordinates of the top intersection point. The z coordinates can be taken from the neighbor intersection point since both bottoms of the struts is in the same level. For strut B, the coordinates of center of the top of the strut can be obtained from MATLAB algorithm and used. However, this part of extraction of the coordinates manually from the algorithm is abandoned since the goal for the project is to find an automated or semi-automated geometry extraction method.

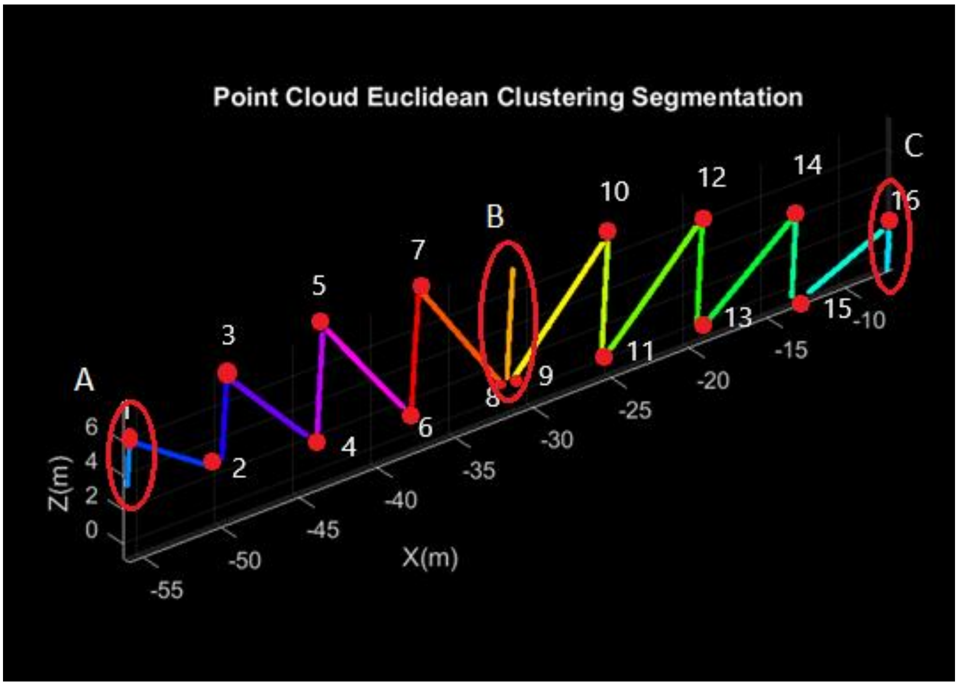

Figure 5.25: Geometry information extraction of the struts based on the points

In addition to the coordinates of two ends of a strut, the radius of each strut can be extracted. The value of 75 mm from the extraction results is chosen as the radius for circular components.

For the I-shaped top chord of the bridge, the coordinates of X insertion points, Y insertion points and Z insertion points have been used as the basis. In addition, the parameters of cross-section of the I-shaped member are used for creation of family in Revit. The cross-section parameters are shown in table 7.

Parameters	Value (m)
Width	0,34
Height	0,40
Flange thickness	0,02
Web thickness	0,02

Table 7: Parameters for the cross-section of the top chord

The parameters of the cross-section of the bottom chord can be used in creation of family in Revit, which are shown in table 8.

Parameters	Value (m)
Width	0,26
Height	0,26
Flange thickness	0,02
Web thickness	0,015

Table 8: Parameters for the cross-section of the bottom chord

Figure 5.26 shows the parameters that can be extracted in order to create the parametric BIM-model of the pedestrian bridge.

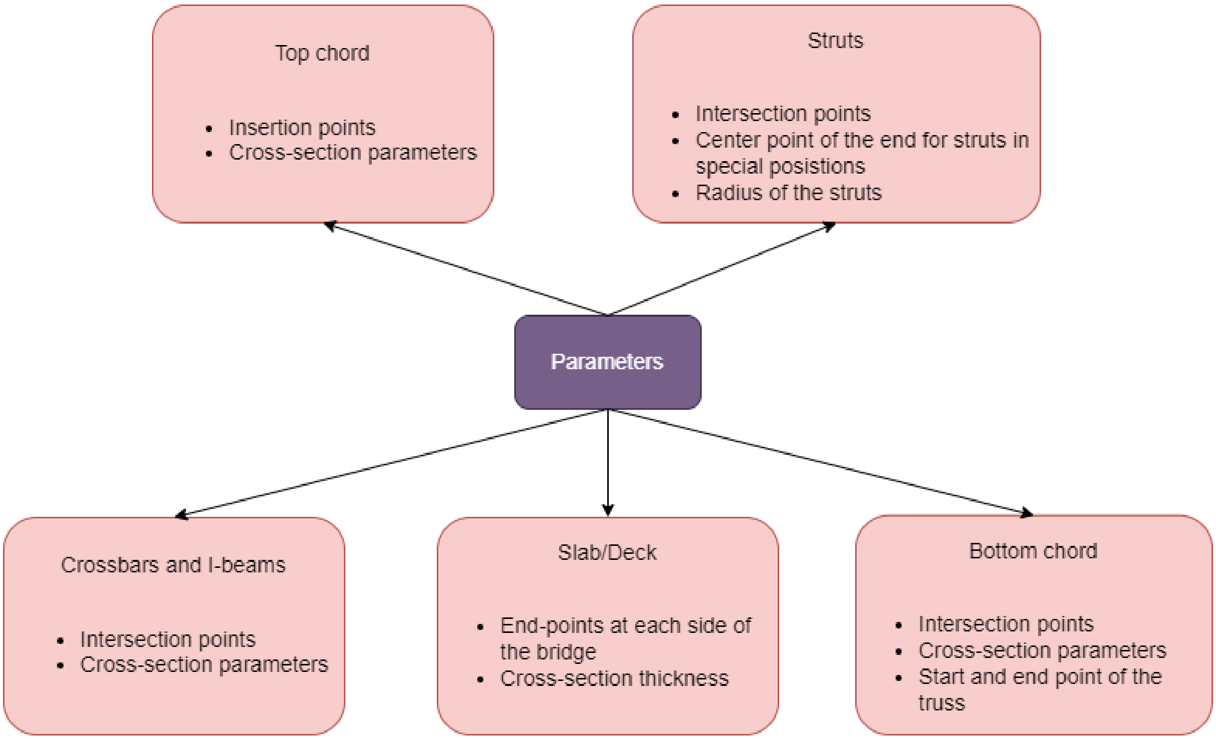

Figure 5.26: Parameters extracted for creation of BIM-model

5.3.2 Creation of the BIM-model

This subchapter defines the parameters that have been used in creation of parametric BIM-model of the pedestrian bridge. These parameters are obtained from geometry information extraction through algorithms and point cloud. In addition, the creation of each component in Dynamo will be described. The sketch of the whole code for Dynamo can be found in appendix part C.

Circular pipe components:

After geometry information extraction in MATLAB, the intersection points in Euclidean Clustering algorithm for both sides can be calculated. The coordinates of points extracted into Dynamo are shown in table 9-10 and figure 5.27 in red.

x(mm)	y(mm)	z(mm)
-55867.455	5579.7529	-671.63086
-55867.455	5579.7529	2668.9072
-49817.482	5579.7529	-671.63086
-49710.217	5579.7529	4621.5553
-43729.08	5579.7529	-811.52344
-43540.543	5579.7529	5753.2501
-37685.001	5579.7529	-1135.6201
-37436.653	5579.7529	6239.9254
-31633.801	5579.7529	-1182.0717
-25267.406	5579.7529	5846.8018
-25454.973	5579.7529	-1279.5238
-19222.662	5579.7529	5098.5107
-19353.035	5579.7529	-1370.5091
-13204.768	5579.7529	3576.8433
-13304.643	5579.7529	-1468.0638
-7103.5032	5579.7529	1397.4609
-7103.5032	5579.7529	-1468.0638

Table 9: Parameters for circular pipe components on left side of the bridge

x(mm)	y(mm)	z(mm)
-55817.986	687.775	1193.359
-55817.986	687.775	4313.402
-49700.703	687.775	1193.359
-49685.085	687.775	6374.069
-43667.049	687.775	839.111
-43488.38	687.775	7288.822
-37631.111	687.775	1088.074
-37379.28	687.775	7919.193
-31483.343	687.775	537.354
-25241.268	687.775	7407.471
-25392.426	687.775	989.567
-19149.042	687.775	7171.57
-19352.43	687.775	292.967
-13140.738	687.775	5444.58
-13205.838	687.775	235.873
-13205.838	687.775	235.873
-7059.258	687.775	3254.883
-7059.258	687.775	235.873

Table 10: Parameters for circular pipe components on right side of the bridge

The first and last points in each table represents the bottom coordinates of the first and last strut at each side. The x- and y-coordinates of these points are taken from the intersection point coordinates of the top of the struts. The z coordinates are the coordinates of neighbor intersection point at the same level.

As shown in figure 5.27, the points a and b have the same x- and y-coordinates. The z-coordinates are the same for point b and c. Hence, the coordinates of point b can be calculated as (x_a , y_a , z_c). The same calculation can be applied to the other end of the bridge, and coordinates of point n can be calculated as (x_m , y_m , z_o).

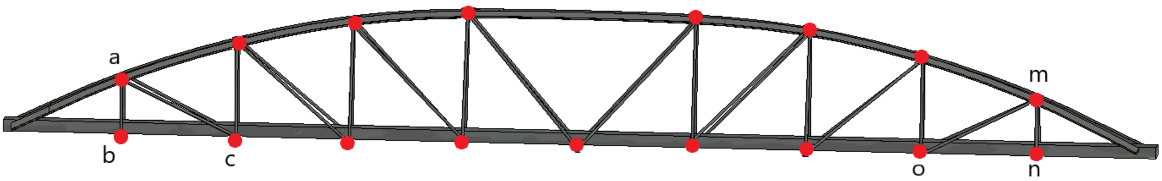

Figure 5.27: Points extracted from MATLAB into Dynamo

Since the circular pipe components are rotated in Geomagic Control X to deal with the algorithm, the components are rotated back in Dynamo into the original position. The left side is rotated by 15,5 degrees on x-axis and by 11,9 degrees on z-axis. The right side of the bridge is rotated by -15,8 degrees on x-axis and by 12,7 degrees on z-axis.

The radius extracted from MATLAB is set to be 75 mm. The circular pipe components are designed in Dynamo by a node called “Cylinder.ByPointsRadius” which uses length between two points and the radius to extrude a cylinder solid. The node “Solid.Union” is then used to join neighbor cylinder solids. Finally, the custom package node “Form.ByGeometry” is used to convert the input geometry to a Revit form object.

Top chords:

The first step of creating top chords on both sides is to establish a family in Revit based on parameters from point cloud in Geomagic Control X. This family is created in 2D space and is extruded with the help of custom nodes in Dynamo.

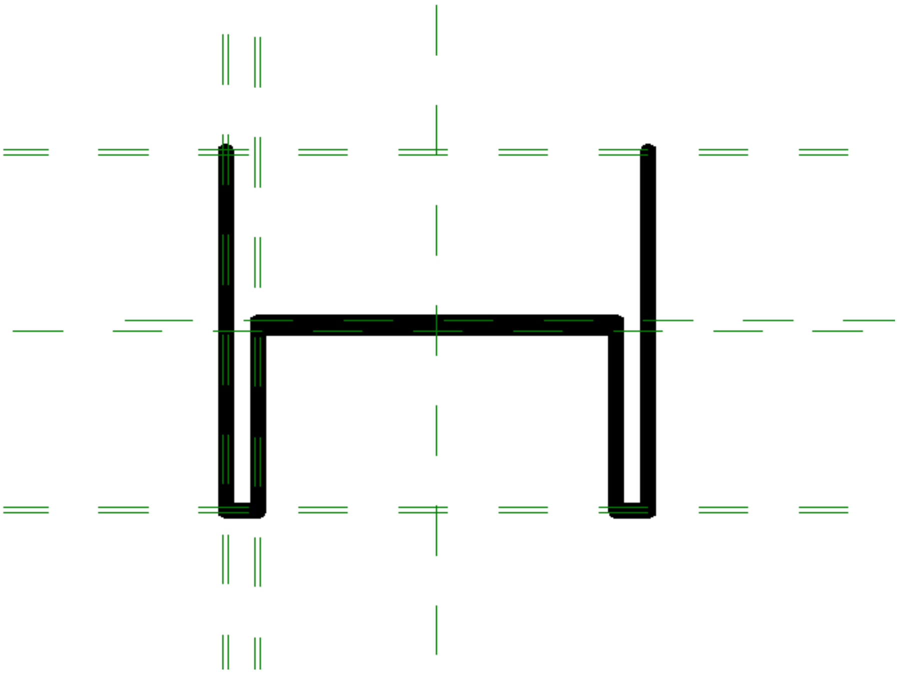

Figure 5.28: Family created in Revit for top chords

The next step is to extract the parameters from polynomial curve fitting algorithm in MATLAB into Excel file. The Excel files are then used as input to Dynamo. The data extracted in this step are the insertion point coordinates of the curve fitted to the top chords. These parameters are shown in table 11-12.

x(mm)	y(mm)	z(mm)
-61733.97	5579.15	-24.68
-59733.97	5579.15	917.53
-57733.97	5579.15	1839.03
-55733.97	5579.15	2653.79
-53733.97	5579.15	3363.35
-51733.97	5579.15	3988.88
-49733.97	5579.15	4543.15
-47733.97	5579.15	5025.30
-45733.97	5579.15	5426.09
-43733.97	5579.15	5736.12
-41733.97	5579.15	5952.46
-39733.97	5579.15	6081.68
-37733.97	5579.15	6139.09
-35733.97	5579.15	6145.01
-33733.97	5579.15	6119.66
-31733.97	5579.15	6077.95
-29733.97	5579.15	6025.84
-27733.97	5579.15	5958.96
-25733.97	5579.15	5863.85
-23733.97	5579.15	5721.58
-21733.97	5579.15	5512.63
-19733.97	5579.15	5221.86
-17733.97	5579.15	4841.94
-15733.97	5579.15	4373.89
-13733.97	5579.15	3823.91
-11733.97	5579.15	3197.01
-9733.97	5579.15	2489.55
-7733.97	5579.15	1685.57
-5733.97	5579.15	764.95
-3733.97	5579.15	-264.03
-1733.97	5579.15	-1289.77

Table 11: Insertion points for the left top chord

x(mm)	y(mm)	z(mm)
-61665.13	683.05	1713.75
-59665.13	683.05	2625.49
-57665.13	683.05	3556.91
-55665.13	683.05	4384.77
-53665.13	683.05	5098.77
-51665.13	683.05	5720.69
-49665.13	683.05	6268.35
-47665.13	683.05	6745.64
-45665.13	683.05	7145.88
-43665.13	683.05	7459.69
-41665.13	683.05	7682.34
-39665.13	683.05	7817.81
-37665.13	683.05	7879.15
-35665.13	683.05	7885.57
-33665.13	683.05	7857.61
-31665.13	683.05	7811.82
-29665.13	683.05	7756.64
-27665.13	683.05	7690.16
-25665.13	683.05	7600.56
-23665.13	683.05	7469.05
-21665.13	683.05	7274.41
-19665.13	683.05	6998.24
-17665.13	683.05	6629.16
-15665.13	683.05	6164.72
-13665.13	683.05	5609.80
-11665.13	683.05	4971.55
-9665.13	683.05	4252.25
-7665.13	683.05	3443.49
-5665.13	683.05	2528.51
-3665.13	683.05	1502.78
-1665.13	683.05	428.24

Table 12: Insertion points for the right top chord

Using the insertion points, a smooth curve can be designed through the points in Dynamo. Using a custom node, the family made in Revit can be placed on the curve. This family can then be extruded along the curve.

Similar to the circular pipe components the left top chord is rotated by 15,5 degrees on x-axis and by 11,9 degrees on z-axis. The right top chord of the bridge is rotated by -15,8 degrees on x-axis and by 12,7 degrees on z-axis. This is to rotate the top chords to the original position.

Finally, the custom package node “Form.ByGeometry” is used to convert the input geometry to a Revit form object.

Other components:

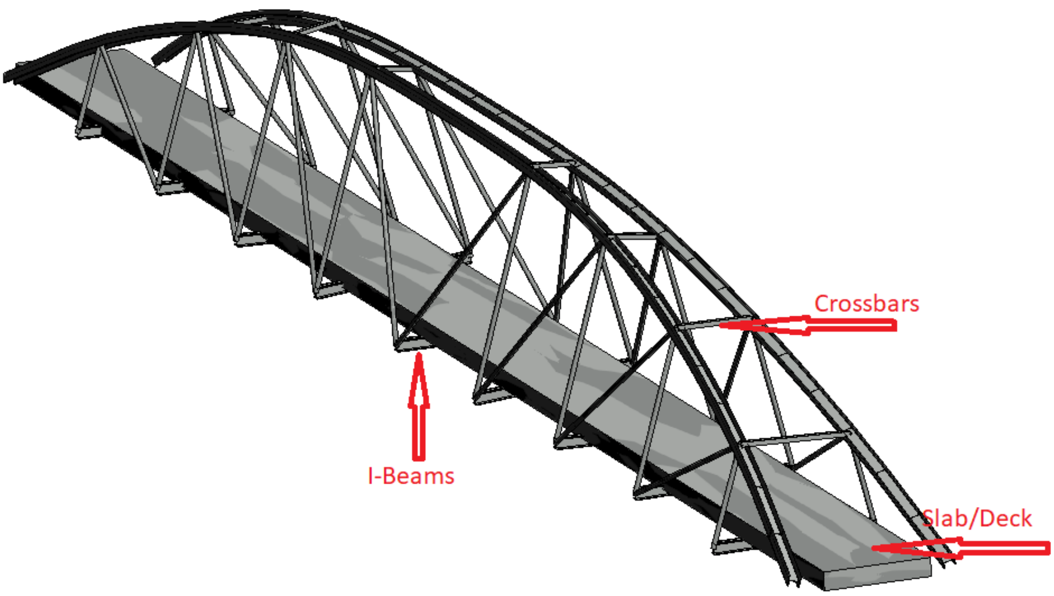

Figure 5.29: Other components to be designed in Dynamo

In order to design the slab of the pedestrian bridge, two points at each end of the bridge are chosen. These points are then connected to each other to create a rectangular form. The boundary of the rectangle is achieved by drawing smooth lines/curves between the points. The node "Surface.ByPatch" is then used to fill the interior of the closed boundary defined by the input curves.

By using the node "Surface.Thicken" the thickness of the slab can be set in. The thickness of the deck can be obtained from point cloud.

For the crossbars, the intersection points at the top from the Euclidean Clustering algorithm for both sides of the bridge have been used. The process of the design in Dynamo is similar to the design of the top chords with I-shaped members. This process contains making family and extruding the family along a smooth curve between two across intersection points.

The same process can be applied to the I-beams under the slab/deck. In this case the bottom intersection points have been used.

At the end, the input geometry of the slab/deck, crossbars and I-beams can be converted to Revit form object using custom package node "Form.ByGeometry".

The final result from Revit model is shown in figure 5.30.

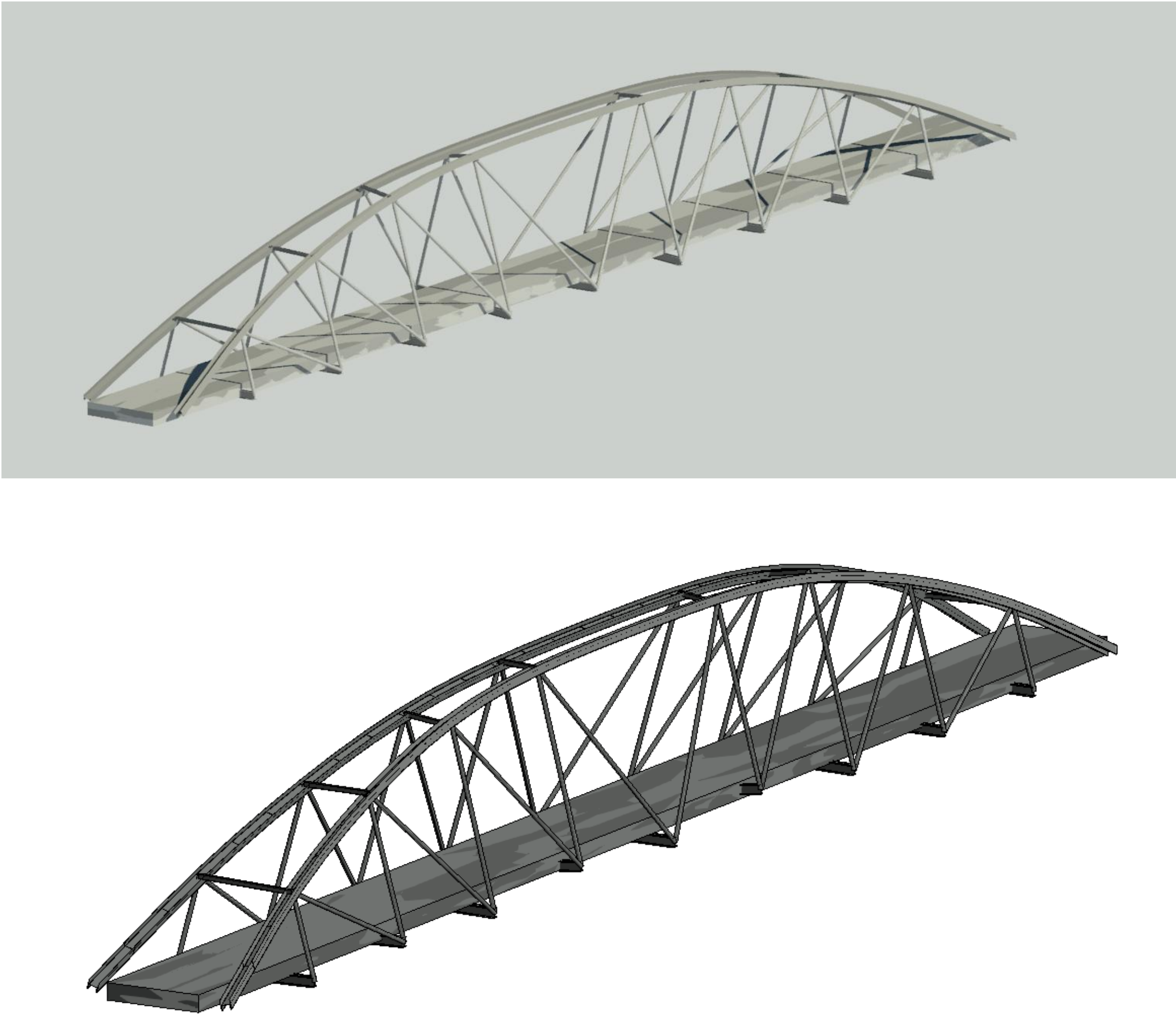

Figure 5.30: The final parametric BIM-model

5.4 Comparison between BIM-model and Point Cloud

It has been performed a 3D deviation analysis of the BIM-model and the point cloud in Geomagic Control X as the last step in the technical route of this thesis. Since the focus is on the curved top chords and the circular pipe components on both sides, the point cloud has been processed and cleaned so that the top chords and the circular pipe components are more visible. The point cloud data is extracted from CloudCompare and imported into Geomagic Control X. The BIM-model is exported as a SAT-file from Revit and imported into Geomagic Control X in the same workspace as the point cloud is. Figure 5.31 shows the overall alignment of the BIM-model and the point cloud.

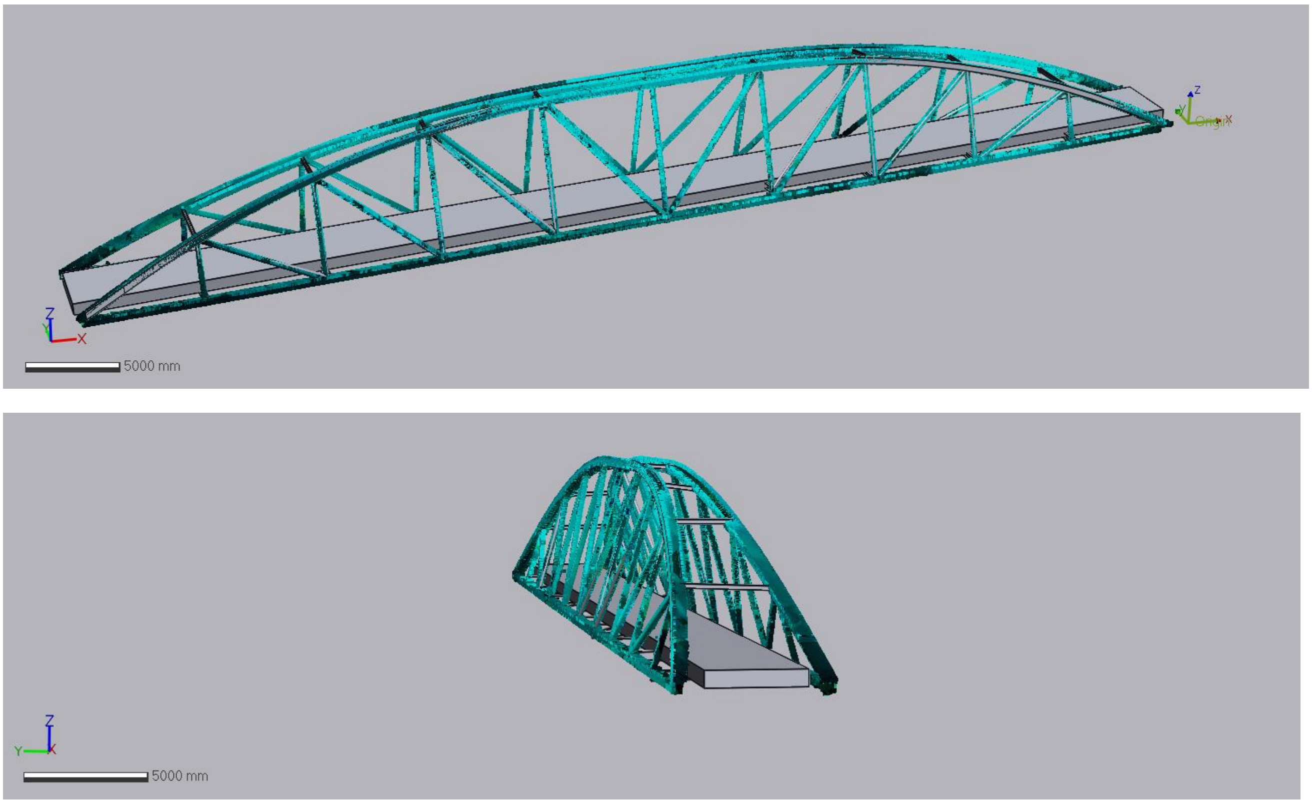

Figure 5.31: The BIM-model and point cloud aligned in Geomagic Control X

After the import of the BIM-model and the point cloud, the "3D Compare" tool has been utilized to measure the deviation between the BIM-model and the point cloud. The "3D Compare" tool creates a deviation colormap that gives a specific color to a certain deviation. Deviations that are in the same range have the same range of color. After the creation of the colormap the "3D compare" tool gives the option to manually pick points to inspect the deviation at these points. It has been chosen a point in every cylinder strut on each side of the bridge and points along the top chords. The points chosen and the deviation of the left and right side of the bridge are shown in figure 5.32 and figure 5.33 respectively.

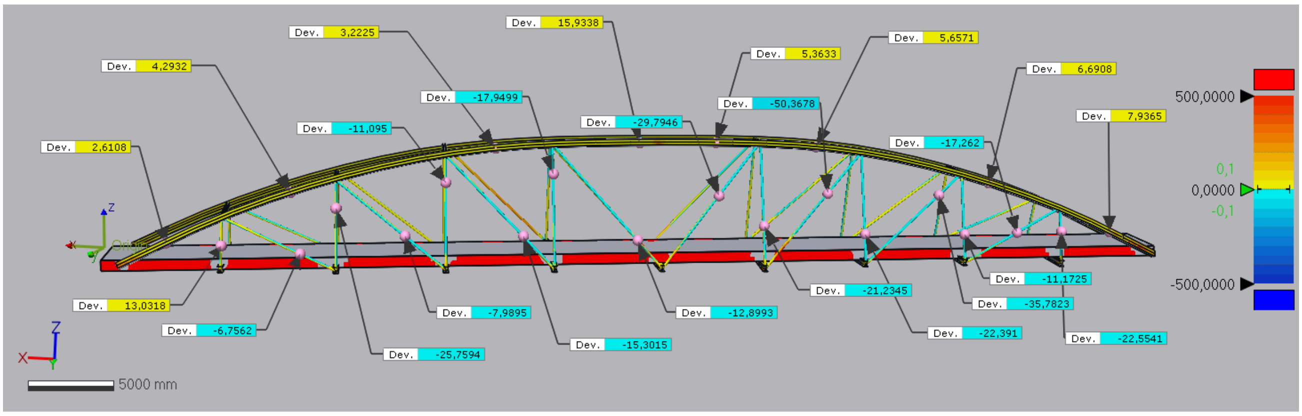

Figure 5.32: Points chosen with corresponding deviation on the left side of the bridge

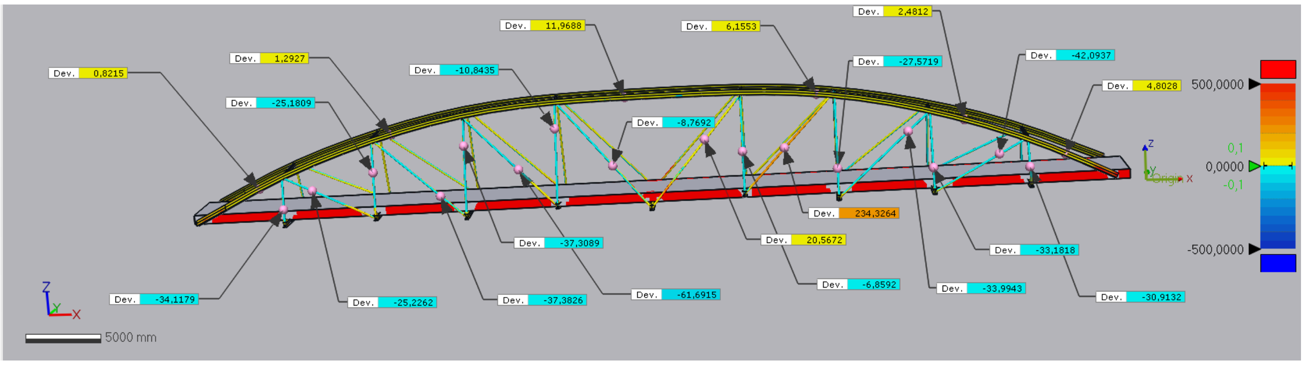

Figure 5.33: Points chosen with corresponding deviation on the right side of the bridge

6 Discussion

Bridges have a crucial role in transport and safety of the public. Having access to a 3D model of a bridge is important for analysis and facility management of a bridge to maintain the safety of the bridge. Most bridges in Norway do not have a 3D model that can be used in analysis. Such a model can help engineers and actors analyze and inspect a bridge structure more efficiently. Long-span spatial curved structures with circular pipe components are widely used in engineering. One of the main problems in the building industry regarding such structures is measuring the overall alignment of this type of structure quickly and accurately. The development in technology and capability of BIM can be utilized to find a solution to this problem.

Based on the situation described above, the authors of the thesis in collaboration with supervisors have attempted to find an answer to the research question below.

"How can the overall alignment of long-span spatial curved structures with circular pipe components be measured quickly and accurately?"

For this purpose, a case study was found. The bridge under research in this thesis is a pedestrian bridge located in the center of Oslo city in Norway. The bridge consists of curved I-shaped steel member on the top, circular pipe steel components as struts, I-shaped steel member bottom chord, steel I-beams under slab and slab made of concrete.

By means of the technology of 3D laser scanning, it is attempted to build a parametric driven BIM model of the existing bridge as a digital twin. In this chapter, various stages of the technical route of the thesis will be discussed, and a conclusion based on the research question will be presented.

The technical route and method applied in this thesis is about finding a long-span spatial curved structure with circular pipe components, scan the structure, make point clouds, apply algorithms to the point clouds for geometry information extraction and build a parametric model of the information obtained.

The results from 3D laser scanning shows to have a good level of accuracy. However, the best results can be achieved through accurate positioning of 3D laser scanner with correct angle to the object to be scanned. To avoid occlusion and capture the whole surface of an object, the scanning should be done from various angles as well. The pedestrian bridge in this thesis is located next to Oslo central station with a lot of traffic from pedestrians and vehicles. To solve this problem, the amount of scanning's from all possible angles is increased which resulted in 14 scans. The results are good enough, but there are some surfaces and components that do not have enough points captured. The improvement of results in this step was abandoned since it is impossible to find a time where the bridge has little amount of traffic. The data collection phase of the technical route of the project can be considered as successful.

The next step in the method applied in this thesis is data processing. The point cloud processing including point cloud registration, merging and initial noise reduction is done in Autodesk Recap. After initial denoising of the point cloud, the point cloud is segmented manually before the input into MATLAB. The components under focus in this research are the curved I-shaped steel members in top chords and circular pipe steel components at both sides of the bridge. Since the bridge is curved in two directions, the point cloud needs to be rotated so that the segmented components of the bridge are parallel to only one plan. The plan chosen in this case is xz plane. For this purpose, the point cloud is exported to Geomagic Control X to deal with the rotation. The rotation of each side of the bridge is as below:

- Left side of the bridge: Rotated by -15,5 degrees on x-axis and by -11,9 degrees on z-axis
- Right side of the bridge: Rotated by +15,8 degrees on x-axis and by -12,7 degrees on z-axis

It seems that both sides are parallel to xz plane after the rotation. However, the adjustments regarding the rotation are done manually with inspection. The rotation of the bridge may affect the exact coordinates of points in point cloud. The main problem is that the structure is curved in two directions due to architectural perspective. With a structure that is curved in only one direction, there would be no need to rotate the point cloud by self-defined angles. Hence, the accuracy of coordinates before input into MATLAB and algorithms would increase. The data processing phase can also be considered as successful since the results obtained are good enough to proceed the application of the method in this thesis.

In geometry information extraction phase of the project, two algorithms are developed to deal with the information extraction. These algorithms are applied to the components under research in this thesis. The algorithm Polynomial Curve Fitting is applied to the top chords of the bridge on each side separately. The algorithm shows to work optimally to extract the coordinates of insertion points which are used in creation of the BIM model in Dynamo. This step may be affected by the rotation done in Geomagic Control X. However, the results obtained from this algorithm seems to be reliable.

The algorithm Euclidean Clustering is applied to circular pipe components on each side of the bridge separately. The algorithm depends highly on a dense point cloud with enough number of points in each cylinder component. The idea is to find the centerline of each circular component which defines the intersection points. Enough points help the algorithm to recognize clusters more easily. After several runs, it seems that there is lack of enough points in some components as a consequence of occlusions in 3D laser scanning. Another factor that is crucial in the developed algorithm are the minimum Euclidean distance between two distinct cluster points and maximum distance from the point to the cylinder surface, which respectively are 0,06 and 0,005. In some of the runs, it was tried to delete the places in point cloud with not enough points, and adjust the minimum Euclidean distance, but it resulted in inaccuracies regarding recognition of clusters, radius, and intersection points. The maximum distance from a point to a cylinder surface depends on the level of accuracy in denoising process. With an accurate denoising in details, this distance can be decreased or even neglected. A dense point cloud with enough points which is denoised accurately and in details will help the algorithm recognize the clusters more easily, which leads to more accurate geometry information extraction of the circular pipe components. The majority of results obtained from this phase of the project are reliable and manageable, and the phase geometry information extraction can be considered as successful.

The next step is to create a parametric BIM model based on the extracted data. The extracted data for the top chords through Polynomial Curve Fitting algorithm in form of insertion points are utilized in Dynamo to design the top chords with a smooth curve. An issue which is important to mention here is that the middle of the top chord flattens. The algorithm extracts best results for a curved structure which is curved whole the way. However, it is tried to fit the best curve, which resulted in a polynomial of 10^{th} degree. With a totally curved structure, the results for parameters extracted would be in a higher level of accuracy. The top chords are however designed successfully in Dynamo by insertion points and families based on parameters from the point cloud. For circular pipe components, the radius of cylinders and intersection points obtained from geometry information extraction has been utilized to design the parametric model. The model of cylinders in Dynamo depends on availability of two points (intersection points) to create a smooth curve as the path for a cylinder. The algorithm cannot define and extract the bottom points for the first and last strut, as well as the top point for the strut in the middle as shown in figure 6.1.

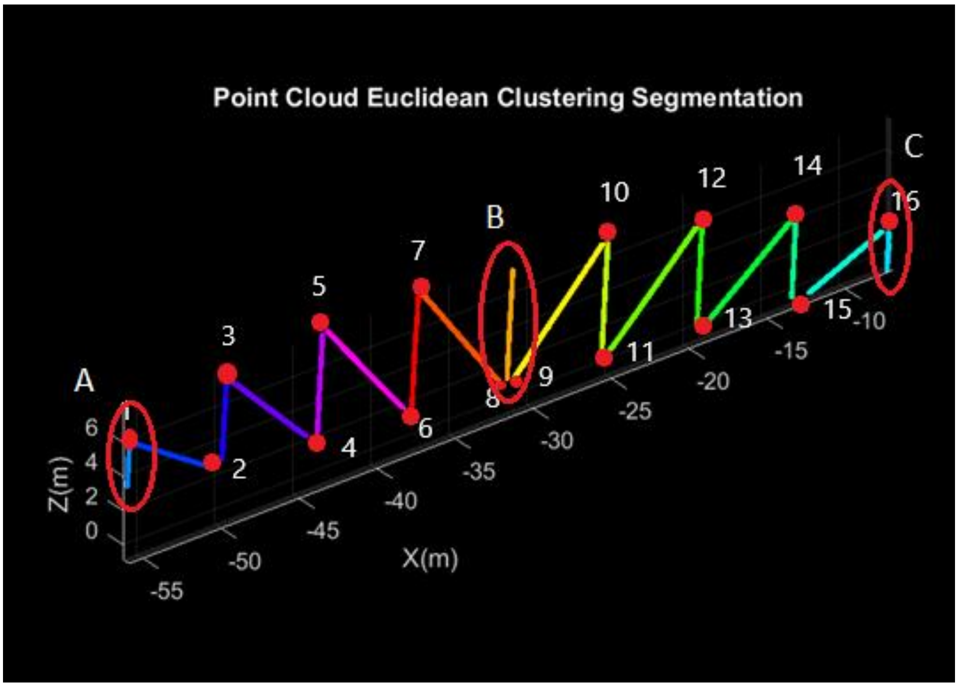

Figure 6.1: Intersection points and struts

There are some results from the algorithm that can be used to find the coordinates of these points. For the strut in the middle, the center point of the upper end of the strut combined with an intersection point in the bottom could be used as two points needed in Dynamo. The problem is that the strut was cut in manual segmentation step. The consequence of this operation is that the real position of the top point cannot be found. Since the goal for the project is to find a way that measures the overall alignment automatically to increase the quickness and effectivity, the design of this strut in Dynamo is abandoned.

The bottom points for the first and last strut can be assumed from the geometry extraction information as described in chapter 5.3.2. The difference between these struts and the strut in the middle is that there is room for assumption regarding the first and last strut. Hence, these struts are included in design of the parametric model as shown in figure 6.2.

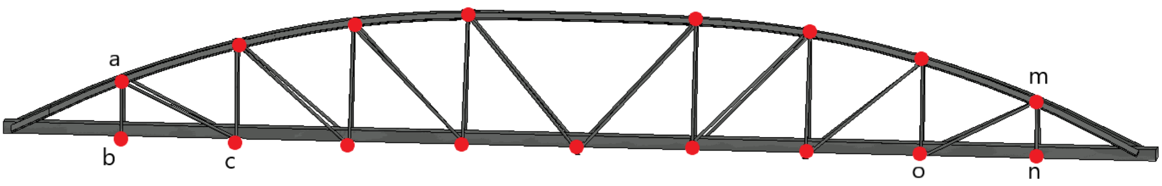

Figure 6.2: The struts modelled in Dynamo using the red points

Other components of the bridge can also be modeled in Dynamo with the information extracted. One way to design the bottom chord is to use the intersection points at the bottom obtained from Euclidean Clustering algorithm combined with the first and last point of the top chord obtained from Polynomial Curve Fitting algorithm as shown in figure 6.3.

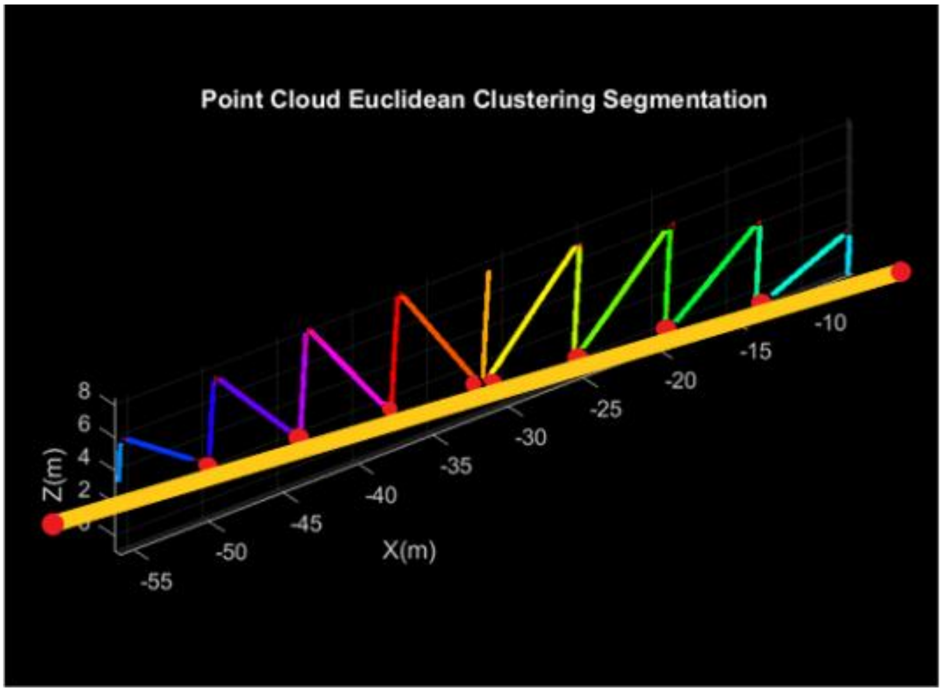

Figure 6.3: The points that can be used in modeling the bottom chord

Since this operation leads to some inaccuracies and goes away from the goal for the project, the design of bottom chords is abandoned.

For the sake of design, the cross bars at the top and the slab with I-beams under it are also modeled in Dynamo. For crossbars at the top, the intersection points across are used. The intersection points across at the bottom are used to model the I-beams. It is not possible to find the necessary points for modeling of all I-beams through the geometry information extraction done in this project. Since the focus of the project is on the I-shaped top chords and circular pipe components of both sides, the creation of parametric BIM model step can be considered as successful.

The last step in the method applied in this thesis is to compare and analyze the deviation between the created BIM-model and the point cloud of the bridge. The comparison objects are top chords and circular pipe components on both sides of the bridge. As shown in figure 6.4, the overall alignment of the components compared seems to be good and every component in the model fall upon the corresponding point cloud components.

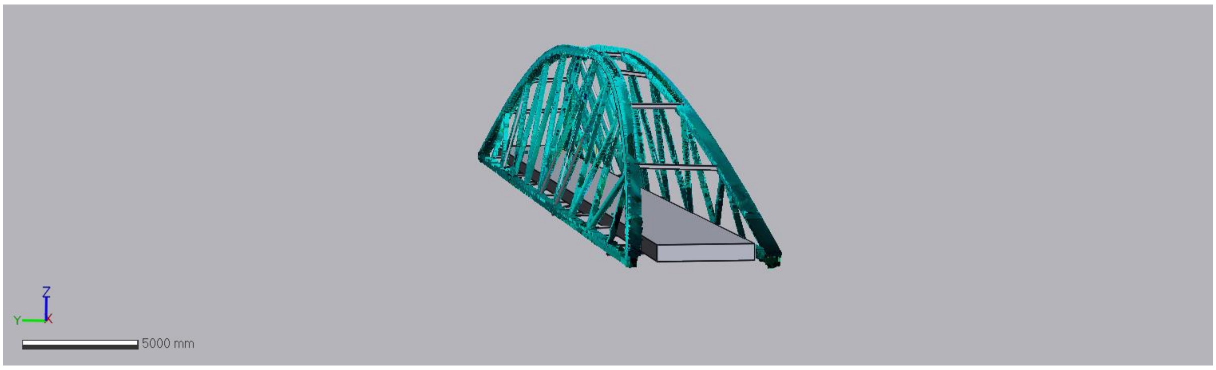

Figure 6.4: Overall alignment of the BIM-model and point cloud

Results from deviation analysis of the top chords show that the deviation range is between 2 mm and 16 mm for the left top chord of the bridge. The deviation is evenly distributed along the top chord as shown in figure 6.5.

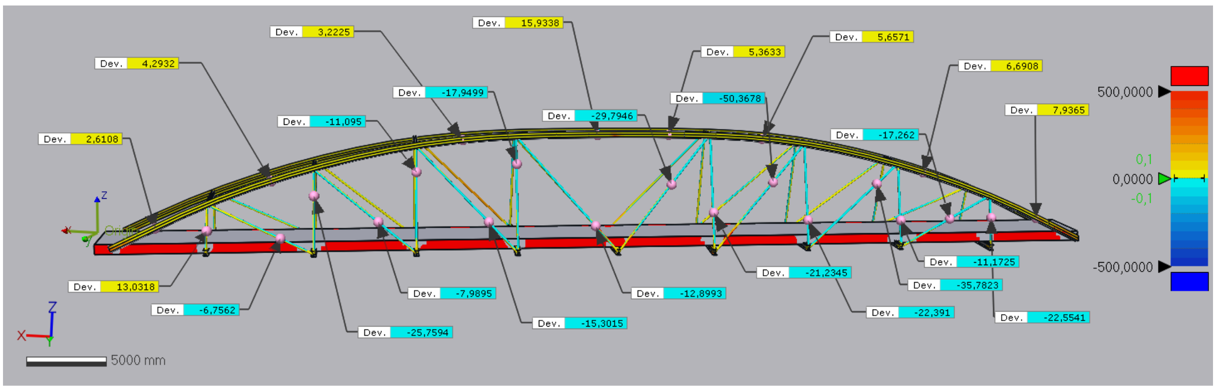

Figure 6.5: Deviation analysis of left side of the bridge

The effect of flattened middle part of the top chord can be seen in the deviation analysis. The algorithm used for geometry information extraction of the top chord in Polynomial Curved

Fitting. The trend and pattern of the outer sides of the curved top chord are more curve-like than linear. The deviation at the flattened part of the top chord is more than the other parts. With a totally curved top chord, the deviation would more likely become less. However, various degrees of polynomial have been tested and applied to the algorithm, and the polynomial degree of 10^{th} order gives reliable results.

For the circular pipe components on the left side, the deviation is in a range between 6 mm and 50 mm. The deviation is generally evenly distributed. The deviations for top chord and circular pipe components on right side of the bridge are shown in figure 6.6.

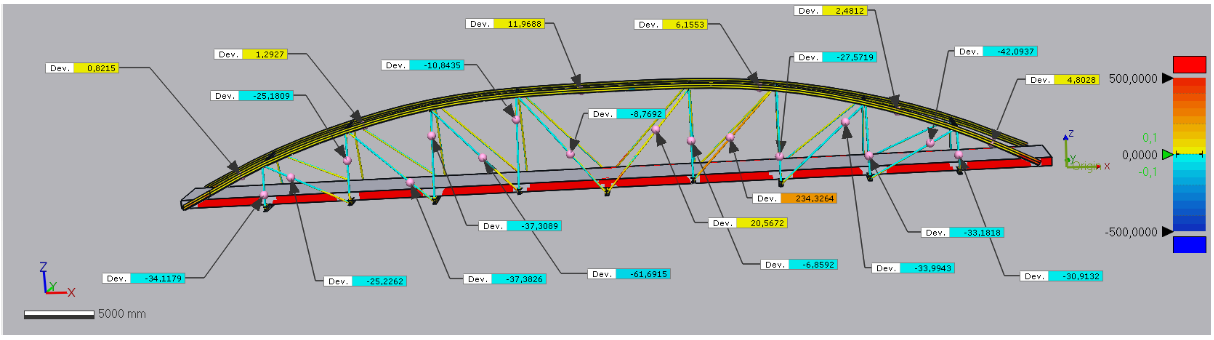

Figure 6.6: Deviation analysis of right side of the bridge

The top chord of the right side has a deviation between 0 mm and 12 mm with the maximum deviation in the flattened part of the top chord similar to the left side of the bridge.

The deviation for circular pipe components on right side has a range from 8 mm to 62 mm in general. However, there is a sudden change in deviation for one of the components where the deviation is around 234 mm. This component is the same component that earlier got warnings in the algorithm. The warning was about the lack of enough points in the point cloud for the component. When there are not enough points in point cloud, the algorithm will not be able to recognize the clusters and find the centerline of the clusters. For this component, the point cloud had a lower density at the lower part of the strut. To solve this problem, the strut was cut a bit more so that the part with a higher density can be run through the algorithm. This operation resulted in an optimal geometry information extraction in general for this side of the bridge, but the sudden deviation for this component was expected.

In general, with a less complex bridge with no need for rotation before applying the algorithms, and with an accurate and dense point cloud without occlusion, the results would more likely be in a much higher level of accuracy.

7 Conclusion

Measuring the overall alignment of long-span spatial curved structures with circular pipe components quickly and accurately is an important aspect that increase the effectiveness of building digital twins.

The research question in this thesis is as below:

"How can the overall alignment of long-span spatial curved structures with circular pipe components be measured quickly and accurately?"

The overall alignment of long-span spatial curved structures with circular pipe components can be measured through the method described in this thesis quickly and in an effective way.

This thesis has an approach to the research question through the technical route and method applied on a real case study. The thesis contains a method to automatically build an as-build BIM model of the existing bridge based on automatic geometry information extraction of components in MATLAB. The automation of this process increases the quickness of the work.

3D laser scanning is an accurate and effective way of collecting data from an existing structure.

The data collecting from 3D laser scanning can be used in the algorithms in MATLAB to extract parameters and geometry information automatically. The extracted parameters can be used in Dynamo to construct an as-built BIM model of the existing structure.

The results obtained in this thesis has a good level of accuracy, but there are still some ways of improving the results. One way is to increase the accuracy and density of the point cloud through good planning before the 3D laser scanning. In this way the recognition of components in algorithms will be easier, and the accuracy of the geometry information extraction increases. Another way is to deal with the algorithms in a way that there would be no need to rotate the structure manually.

All things considered, the case study examination in this thesis was successful, and it can be concluded that the method applied in this thesis can help to measure the overall alignment of long-span curved structures with circular pipe components quickly and accurately and paves the way for smarter and more efficient AEC-industry.

Further Work

It is recommended for further work that one extensively explores the field of 3D laser scanning and programming before attempting to implement this method as it requires sufficient knowledge and experience in these fields.

It is also recommended that that the method and technical route of this thesis is applied to other cases in order to get an overview on whether this method and technical route can be considered case-specific or not.

It is already known from this thesis that the algorithm works best when the structure is curved in one direction. The case structure in this thesis is rotated manually so that it is parallel with the XZ-plane which may affect the results. It is therefore recommended for future work to develop an algorithm that works without the need to rotate the structures. In addition, for future work a more complex algorithm can be developed in order to get more accurate and reliable data even with point clouds that do not have enough points and density.

The components that have been through the automatic geometry information extraction in this thesis are curved steel member and circular pipe components. The future work can also contain developing algorithms for geometry information extraction of other components in a bridge.

Sources

1. Fogg, J. *Bridge Inspection is more important than you think*. 2015; Available from: https://www.linkedin.com/pulse/bridge-inspection-more-important-than-you-think-jason-fogg.
2. Statsbygg. *Hva er BIM?* 2022; Available from: https://go.nanolearning.com/LessonViewer/?da=1961473&key=EvXxndoAgqhKVwWu&mode=1&du=10841782.
3. Technology, Z.G.I.o., *State of BIM adoption across Europe (Maps)*. 2019: https://www.e-zigurat.com/blog/en/state-of-bim-adoption-europe/.
4. Autodesk, *Building Information Modeling*. 2002: http://www.laiserin.com/features/bim/autodesk_bim.pdf.
5. Eastman, C., Teicholz, P., Sacks, R. & Lee, G., *BIM Handbook: A Guide to Building Information Modeling for Owners, Designers, Engineers, Contractors, and Facility Managers, Third Edition*. 2018: https://onlinelibrary-wiley-com.ezproxy.oslomet.no/doi/book/10.1002/9781119287568.
6. Wikipedia. *Parametric design*. 2021; Available from: https://en.wikipedia.org/wiki/Parametric_design.
7. MCH. *WHAT IS PARAMETRIC DESIGN?* 2021; Available from: https://www.mchmaster.com/news/what-is-parametric-design/.
8. Phillips, S.M. *Parametric Design : a Brief History*. 2010.
9. *Robotic Form-Finding and Construction Based on the Architectural Projection Logic*. 2017: https://www.researchgate.net/publication/318103333_Robotic_Form-Finding_and_Construction_Based_on_the_Architectural_Projection_Logic.
10. *What is Parametric Design?* 2022: https://robazzo.com/journal/what-is-parametric-design.
11. Bilberg, A., Malik, A.A., *Digital twin driven human–robot collaborative assembly.* 2019.
12. *Digital Twins Standards Following an Open Source Approach*. 2021: https://www.fiware.org/events/digital-twins-standards-following-an-open-source-approach/.

13. Autodesk. *Tverrfaglig BIM-programvare for koordinert design av høy kvalitet*. 2021; Available from: https://www.autodesk.no/products/revit/overview?term=1-YEAR&tab=subscription.
14. Somani, N., *Things that Revit and BIM can do together!* 2019: https://knowledge.autodesk.com/search-result/caas/simplecontent/content/things-revit-and-bim-can-do-together.html.
15. Autodesk. *Reality capture and 3D scanning software for intelligent model creation*. 2021; Available from: https://www.autodesk.com/products/recap/overview?term=1-YEAR&tab=subscription.
16. Dynamo. *What is Dynamo?* 2021; Available from: https://primer.dynamobim.org/01_Introduction/1-2_what_is_dynamo.html.
17. Software, A. *What is Autodesk Dynamo?* 2022; Available from: https://www.asti.com/ultimate-guide-to-autodesk-dynamo/.
18. MathWorks. *Explore the wide range of product capabilities, and find the solution that is right for your application or industry.* 2022; Available from: https://www.mathworks.com/solutions.html#applications.
19. CloudCompare. *CloudCompare*. 2022; Available from: https://www.danielgm.net/cc/.
20. Systems, D. *Geomagic Control X*. 2022; Available from: https://www.3dsystems.com/software/geomagic-control-x.
21. Geo-matching. *Airborne Laser Scanning*. 2022; Available from: https://geo-matching.com/airborne-laser-scanning.
22. Vandrie, R., *ANUGA:-the FREE Ocean Impact model*. 2009: https://www.researchgate.net/publication/309683426_ANUGA-the_FREE_Ocean_Impact_model.
23. Topo, P. *The Differences Between Terrestrial and Mobile Laser Scanning*. 2020; Available from: https://blog.topodot.com/differences-between-mobile-laser-scanning-and-terrestrial-laser-scanning/.
24. Shumilov, B., *A Study on Modeling of Road Pavements Based on Laser Scanned Data and a Novel Type of Approximating Hermite Wavelets*. 2015: https://www.researchgate.net/publication/281650909_A_Study_on_Modeling_of_Road_Pavements_Based_on_Laser_Scanned_Data_and_a_Novel_Type_of_Approximating_Hermite_Wavelets/figures?lo=1.
25. Liu Yang, J.C.P.C., Qian Wang, *Semi-automated generation of parametric BIM for steel structures based on terrestrial laser scanning data.* 2020.

26. Jickling, R., *Advantages of 3D Terrestrial Laser Scanners*. 2020: https://hivevp.com/advantages-of-3d-terrestrial-laser-scanners/.
27. *Large laser scanner checkerboard targets as a set*. 2022: https://shop.laserscanning-europe.com/Large-laser-scanner-checkerboard-targets-as-a-set.
28. Depot, T.H., *5.7 in. Laser Point Cloud Scan Reference Target Sphere with Magnetic Base (6-Pack)*. 2021: https://www.homedepot.com/p/AdirPro-5-7-in-Laser-Point-Cloud-Scan-Reference-Target-Sphere-with-Magnetic-Base-6-Pack-707-6-M/313895060.
29. Thilmany, J. *What Is a Point Cloud?* 2018; Available from: https://constructible.trimble.com/construction-industry/what-is-a-point-cloud.
30. M.V.Jerabandi. *MODULE 2: Branching and Looping*. Available from: http://rechulkoti.edu.in/wp-content/uploads/2018/04/MODULE_2_Decision_Making_and_Looping.pdf.
31. GeeksforGeeks. *Introduction to Algorithms*. 2022; Available from: https://www.geeksforgeeks.org/introduction-to-algorithms/?ref=lbp.
32. Education, I.C. *Unsupervised Learning*. 2020; Available from: https://www.ibm.com/cloud/learn/unsupervised-learning.
33. github-wiki. *Euclidean clustering*. 2022; Available from: https://github-wiki-see.page/m/cse481sp17/cse481c/wiki/Lab-32:-Euclidean-clustering#:~:text=Euclidean%20clustering%20essentially%20groups%20points,that%20a%20cluster%20can%20contain.
34. Qualtrics. *What is cluster analysis? When should you use it for your survey results?* 2022; Available from: https://www.qualtrics.com/experience-management/research/cluster-analysis/.
35. Zach, *Curve Fitting in Python (With Examples)*. 2021: https://www.statology.org/curve-fitting-python/.
36. Glen, S. *Curve Fitting*. 2018; Available from: https://www.statisticshowto.com/curve-fitting/.
37. Kenton, W. *Least Squares Method*. 2022; Available from: https://www.investopedia.com/terms/l/least-squares-method.asp.
38. OriginLab, *Curve and Surface Fitting*. 2022: https://www.originlab.com/index.aspx?go=Products/Origin/DataAnalysis/CurveFitting.

39. Moore, J. *An Introduction to Sliding Window Algorithms*. 2019; Available from: https://levelup.gitconnected.com/an-introduction-to-sliding-window-algorithms-5533c4fe1cc7.
40. Topcon. *Compact and lightweight multi-adjustable laser scanner*. 2020; Available from: https://www.topconpositioning.com/scanning/laser-scanners/gls-2000.
41. Topcon, *Compact and lightweight multi-adjustable laser scanner*. 2022: https://www.topconpositioning.com/scanning/laser-scanners/gls-2000.
42. Topcon, *Instruction Manual- LASER SCANNER GLS 2000 series*. 2022: https://www.manualslib.com/manual/1571548/Topcon-Gls-2000-Series.html?page=15#manual.

Appendix – Part A

```
clear

webPoint=load('polyleft.txt');
webPoint=webPoint(:,1:3);
fitOrder=polyfit(webPoint(:,1),webPoint(:,3),10);
fitValue=polyval(fitOrder,webPoint(:,1))
D_value=fitValue-webPoint(:,3);
stdDeviation=std(D_value)

figure(1)
scatter(webPoint(:,1),webPoint(:,3),'ob')
hold on
scatter(webPoint(:,1),fitValue,'or')

d=2;
X_insertionPoint = min(webPoint(:,1)):d:max(webPoint(:,1))
fitOrder_derivative=polyder(fitOrder);
Z_insertionPoint=polyval(fitOrder,X_insertionPoint);
K_insertionPoint=polyval(fitOrder_derivative,X_insertionPoint);
radian_insertionPoint=atan(K_insertionPoint);
```

```
for i=1:length(X_insertionPoint)

    Rxz{i}= [cos(radian_insertionPoint(i)) 0 -sin(radian_insertionPoint(i))
                        0              1                0
            sin(radian_insertionPoint(i)) 0 cos(radian_insertionPoint(i))];

    webPoint_Transform=(webPoint-[X_insertionPoint(i) 0 Z_insertionPoint(i)])*Rxz{i};


    webPoint_d=webPoint_Transform(find(webPoint_Transform(:,1)<d&webPoint_Transform(:,1)>-d),:);

    Z_webPoint_d=mean(webPoint_d(:,3));
    Y_webPoint_d=mean(webPoint(:,2));


    Rxzn{i}= [cos(-radian_insertionPoint(i)) 0 -sin(-radian_insertionPoint(i))
                         0             1                0
            sin(-radian_insertionPoint(i)) 0 cos(-radian_insertionPoint(i))];

    webPoint_dInverse=webPoint_d*Rxzn{i}+[X_insertionPoint(i) 0 Z_insertionPoint(i)];

    linear(i,:)=[0 0 Z_webPoint_d]*Rxzn{i}+[X_insertionPoint(i) Y_webPoint_d Z_insertionPoint(i)];
```

```
    color={'go','ro','co','mo','yo','ko'};
    figure(4);
    plot3(webPoint_dInverse(:,1),webPoint_dInverse(:,2),webPoint_dInverse(:,3),color{mod(i,6)+1},'MarkerSize',1);
    hold on
    plot3(X_insertionPoint,mean(webPoint(:,2))*ones(size(X_insertionPoint,2),1),Z_insertionPoint,'ro--','MarkerSize',10,'MarkerFaceColor','r')

    xlabel('X axis')
    ylabel('Y axis')
    zlabel('Z axis')
    set(gca,'linewidth',2,'fontsize',30,'fontname','Times');
    axis equal

    clear webPoint_Transform  webPoint_d
end

figure(5)
plot(linear(:,1),linear(:,3),'o-r','linewidth',2,'markersize',6);
hold on
plot(X_insertionPoint,Z_insertionPoint,'*k','MarkerSize',5);
hold on
plot(webPoint(:,1),webPoint(:,3),'*b','MarkerSize',2)
```

Appendix – Part B

```
clc;
clear all;

bridgetest = pcread('05 bridgetestCylinder - Cloud.pcd');

pcshow(bridgetest);
title('pcshow(bridgetest) Example');

minDistance = 0.06;

[labels,numClusters] = pcsegdist(bridgetest,minDistance);


figure(1);
colormap(hsv(numClusters));
pcshow(bridgetest.Location,labels);
title('Point Cloud Euclidean Clustering Segmentation');
xlabel('X(m)');
ylabel('Y(m)');
zlabel('Z(m)');
```

```
k=1;
for i=1:max(labels(:,1))

    bridgetestCylinder=bridgetest.Location(find(labels(:,1)==i),:);
    ptCloud = pointCloud(bridgetestCylinder(:,1:3));
    pcwrite(ptCloud, 'test.pcd', 'Encoding', 'ascii');
    bridgetestCylinder_pc = pcread('test.pcd');
    maxDistance = 0.005;


    bridgetestCylinderMean=[mean(bridgetestCylinder(:,1)),mean(bridgetestCylinder(:,2)),mean(bridgetestCylinder(:,3))];
    centeredLine=bsxfun(@minus,bridgetestCylinder,bridgetestCylinderMean);
    [U,S,V]=svd(centeredLine);
    referenceVector=V(:,1);

    model = pcfitcylinder(bridgetestCylinder_pc,maxDistance,referenceVector);

    if isempty(model)==1;
        continue
    end

    modelgg{k,1}=model;
    modelRadius(k,1)=modelgg{k, 1}.Radius;
    modelCenter_Orientation(k,1:6)=[modelgg{k, 1}.Center,modelgg{k, 1}.Orientation];

    k=k+1;
    figure(2);


    colormap(hsv(numClusters));
    pcshow(bridgetest.Location,labels);
    title('Point Cloud Euclidean Clustering Segmentation');
    xlabel('X(m)');
    ylabel('Y(m)');
    zlabel('Z(m)');
    hold on
    plot(model);

end

modelCenter_Orientation0=sortrows(modelCenter_Orientation,1);
modelRadiusMedian=median(modelRadius);
modelRadiusbMean=mean(modelRadius);
figure(2)

colormap(hsv(numClusters));
pcshow(bridgetest.Location,labels);
title('Point Cloud Euclidean Clustering Segmentation')
xlabel('X(m)');
ylabel('Y(m)');
zlabel('Z(m)');
hold on
plot(model)


k=modelCenter_Orientation0(:,6)./modelCenter_Orientation0(:,4);
b=modelCenter_Orientation0(:,3)-k.*modelCenter_Orientation0(:,1);
```

```
for i=1:size(modelCenter_Orientation0,1)-1

    k1=k(i,:);
    k2=k(i+1,:);
    b1=b(i,:);
    b2=b(i+1,:);
    intersection(i,1)=(b2-b1)/(k1-k2);
    intersection(i,2)=k2*intersection(i,1)+b2;
    intersection(i,3)=mean(modelCenter_Orientation0(:,2));

end

figure(3)

colormap(hsv(numClusters));
pcshow(bridgetest.Location,labels);
title('Point Cloud Euclidean Clustering Segmentation');
xlabel('X(m)');
ylabel('Y(m)');
zlabel('Z(m)');
hold on
plot3(intersection(:,1),intersection(:,3),intersection(:,2),'r-');
```

Appendix – Part C