# NTNU

Norwegian University of
Science and Technology

# Scan to BIM using Smartphone 3D Scanning

## Erlend Stendahl Spets

Civil and Environmental Engineering
Submission date: June 2018
Supervisor: Tor Guttorm Syvertsen, KT

Norwegian University of Science and Technology
Department of Structural Engineering

| **Department of Structural Engineering** | ACCESSIBILITY |
| Faculty of Engineering | |
| **NTNU- Norwegian University of Science and Technology** | OPEN |

# MASTER THESIS 2018

| SUBJECT AREA: | DATE: | NO. OF PAGES: |
| --- | --- | --- |
| Construction Informatics | 11.06.2018 | 118 |

TITLE:

### Scan to BIM using Smartphone 3D Scanning

Skanning til BIM ved bruk av smarttelefon 3D skanning

BY:

Erlend Stendahl Spets

SUMMARY:

New tools for digital recreation of existing buildings as building information models, based on 3D scanning, are gradually being adopted by the construction industry. These technologies are emerging as the concept *Scan to BIM.* The use of scanning to gain accurate digital models of reality could revolutionize the way some construction projects are conducted. Today, the scanning is primarily done by using ground fixed laser scanners.

    The aim of this study was to test the possibility of using a smartphone for Scan to BIM projects. To achieve this, research of current state of the art technologies and testing various procedures of scanning, post processing and BIM creation were conducted. Then, a full-scale smartphone Scan to BIM experiment was carried out on a suitable building. The experiment included various aspects of the technology such as procedure, precision and efficiency. This thesis provides a detailed description of the tests leading to the experiment and the established Scan to BIM procedure.

    The experiment proves that it is possible to create a BIM from smartphone scanning. As there is no straightforward method of creating a BIM from this kind of scanning, the process proved to be challenging and several software packages were needed. The smartphone scanning was surprisingly accurate. On the contrary, the final BIM was prone to error as the process of creating the BIM was mainly manual and based on the best fit by visual estimates. Overall, using a smartphone for scanning proved to be an efficient tool, as it is a convenient way of scanning and accessing all parts of a building. The experiment also provided some evidence that this is a faster way of scanning and establishing a BIM than methods currently in use and that smartphone scanning could easily be combined with other scanning techniques.

RESPONSIBLE  TEACHER: Tor Guttorm Syvertsen

SUPERVISOR: Tor Guttorm Syvertsen

CARRIED OUT AT: Department of Structural Engineering, NTNU.

**Department of Structural Engineering**

FACULTY OF ENGINEERING
NTNU – Norwegian University of Science and Technology

# Master Thesis 2018

for

*Erlend Stendahl Spets*

# Scan to BIM using Smartphone 3D Scanning

*Skanning til BIM ved bruk av smarttelefon 3D skanning*

**Background**

BIM is an established technology within the construction industry. 3D geometric models of existing buildings have for some time been created by laser scanning techniques. These models can be refined into BIMs by adding relevant properties. Recent developments of cameras and software for mobile phones have made it feasible to use smartphones for 3D scanning.

**The idea**

Smartphones for 3D scanning of buildings have the potential of making *Scan to BIM* available for a wide range of applications even for small companies in the construction industry and elsewhere. Time has come to explore *Scan to BIM* by smartphone in a realistic situation. The objective is to establish a BIM in Autodesk Revit of an existing building by using a smartphone for scanning.

**Scope of Work**

The assignment comprises:

- Survey and acquisition of appropriate equipment
- Small-scale practising to a skilful level of scanning and post-processing.
- Locating a suitable building for an experimental *Scan to BIM* by smartphone
- Planning and executing a complete, experimental *Scan to BIM* of the building
- Reviewing the experiment and the resulting BIM
- Recommendations for scanning equipment, methodology, software, etc.

**Deliverables**

A report is to be handed in to the Department of Structural Engineering by June 11, 2018.

The thesis may be adjusted due to the progress of work and the interests of the student.

The work is to be organized in accordance with the current instructions.

*Supervisor:* Tor Guttorm Syvertsen

NTNU, 15.01.018

# Preface

This Master's Thesis was written as the final part of the master's program of Civil Engineering at the Norwegian University of Science and Technology in Trondheim. It was written at the Department of Structural Engineering in the period from January to June 2018. The thesis is the result of 20 weeks of work which equals a total of 30 credits.

The topic of this thesis was chosen based on my findings in the project report: *Scan to BIM – A brief introduction*, which was written during the fall of 2017 [1]. The thesis concerns the topic of creation of a Building Information Model (BIM) of an existing building using a smartphone for 3D scanning. This is a relatively recent field of study, meaning that this report will be one of the first research reports concerning this topic.

It is assumed that the reader is familiar with terminology used in the construction industry and has a general understanding of how the industry works. No other previous knowledge is needed, as technology and procedures are thoroughly explained in the report.

The Master's Thesis has been supervised by Professor Tor Guttorm Syvertsen at the Department of Structural Engineering. I would like to express my deepest gratitude towards Prof. Tor, for his guidance during the entire project. Meaningful discussions concerning the project's workflow, writing and the structure of the thesis were highly appreciated.

I would also like to thank Radoslaw Raczynski at NTNU for lending me drone equipment and for his help during drone scanning and drone scan processing. Gratitude is also given to Helge Stendahl for allowing me to use his farmhouse as an object for scanning and for facilitation during the experiment. A last gratitude is given to Cathrine Ro Heuch for highly appreciated assistance during the final stages of the assignment.

<br>

<center>

Trondheim, 2018-06-11

<br>

Erlend Stendahl Spets

</center>

# Abstract

New tools for digital recreation of buildings as building information models, based on 3D scanning, are gradually being adopted by the construction industry. These technologies are emerging as the concept *Scan to BIM*. The use of scanning to gain accurate digital models of reality could revolutionize the way some construction projects are conducted. Today, the scanning is primarily done by using ground fixed laser scanners.

The aim of this study was to test the possibility of using a smartphone for Scan to BIM projects. To achieve this, research of current state of the art technologies and testing of various procedures of scanning, post processing and BIM creation were conducted. Then, a full-scale smartphone Scan to BIM experiment was carried out on a suitable building. The experiment included various aspects of the technology such as procedure, precision and efficiency. This thesis provides a detailed description of the tests leading to the experiment and the established Scan to BIM procedure.

The experiment proves that it is possible to create a BIM from smartphone scanning. As there is no straightforward method of creating a BIM from this kind of scanning, the process proved to be challenging and several software packages were needed. The smartphone scanning was surprisingly accurate. On the contrary, the final BIM was prone to error as the process of creating the BIM was mainly manual and based on the best fit by visual estimates. Overall, using a smartphone for scanning proved to be an efficient tool, as it is a convenient way of scanning and accessing all parts of a building. The experiment also implies that this is a faster way of scanning and establishing a BIM than methods currently in use and that smartphone scanning could easily be combined with other scanning techniques.

# Sammendrag

Nye redskaper for digital gjenskaping av eksisterende bygninger i form av bygnings informasjons modeller, basert på 3D skanning, blir gradvis innført i byggebransjen. Disse teknologiene går under betegnelsen *Skann til BIM*. Bruk av skanning til å lage nøyaktige digitale modeller av virkeligheten kan revolusjonere fremgangsmåten på enkelte byggeprosjekter. I dag brukes det i all hovedsak laserskannere til å utføre skanningen.

Målet med denne studien var å teste hvorvidt det er mulig å benytte smarttelefon til prosjekter innen Skann til BIM. For å finne ut av dette måtte først nåværende forskning innen fagfeltet undersøkes. Deretter fulgte en periode med testing av forskjellige metoder for skanning, prosessering og produksjon av en BIM. Til slutt ble det gjennomført et fullskala Skann til BIM-eksperiment på en passende bygning. Eksperimentet inkluderte undersøkelser av forskjellige viktige aspekter ved teknologien, slik som metode, nøyaktighet og effektivitet. Denne rapporten gir en detaljert beskrivelse av både testene som ledet til eksperimentet og den etablerte Skann til BIM-prosedyren.

Eksperimentet beviser at det er mulig å lage en BIM ved bruk av smarttelefonskanning. Fordi det per dags dato ikke finnes noen etablert metode for å lage en BIM ved bruk av smarttelefonskanning, ble prosessen krevende og flere programvarer måtte benyttes. Å bruke smarttelefon til skanning viste seg å være overaskende nøyaktig. Den endelige BIMen viste seg derimot ikke å være like presis, trolig på grunn av at prosessen med å lage en BIM basert på et skann i all hovedsak er manuell og basert på øyemål. Totalt sett viste det seg at bruk av smarttelefon var en effektiv måte å skanne på. Metoden var praktisk og det var enkelt å skanne samtlige deler av bygningen. Resultatet fra eksperimentet antyder at smarttelefonskanning er raskere enn metoder som benyttes i dag og at det enkelt kan kombineres med nåværende skannemetoder.

# Acronyms and abbreviations

[3D] 3-Dimensional

[ADF] Area Description File

[AR] Augmented Reality

[BIM] Building Information Model/Building Information Modelling

[CAD] Computer-Aided Design

[CCD] Charged Coupled Device

[Laser]Light amplification by stimulated emission of radiation

[LIDAR] Laser Imaging, Detection and Ranging

[IMU] Inertial Measurement Unit

[IR] Infrared

[MEP] Mechanical, Electrical and Plumbing

[UAV] Unmanned Aerial Vehicle

# Contents

# Chapter 1

# Introduction

## 1.1 Background

Building information modelling (BIM) is an established technology within the construction industry. Traditionally, BIMs have been created from scratch using BIM software, but emerging technologies are now enabling new methods for BIM generation out of existing buildings. For some time, 3D geometric models have been generated from actual buildings using mainly laser scanning techniques. These models can be refined into BIMs by adding relevant properties. This is often referred to as *Scan to BIM*, and usually comprises the entire process from scanning to post processing and BIM creation.

Recent development of cameras and software for smartphones has made it feasible to use the smartphone itself for 3D scanning. Replacing expensive laser scanning equipment with a smartphone has the potential of easing the Scan to BIM process and making it cheaper. This could make Scan to BIM available for a wider range of applications and make it available for small companies in the construction industry and elsewhere.

There is currently no straightforward method of creating a BIM out of smartphone 3D scanning. The scanning can either be done by basic photogrammetry or by using dedicated smartphone apps. Post processing is also challenging as there exists a wide diversity of file formats and software. Generation of a BIM out of a 3D geometric model is currently a manual or semi-automated process using a BIM software. Presently, there are some algorithms for an automatic BIM generation under development, but so far with not remarkable success.

## 1.2   Objectives

The main objective of this thesis is to determine if it is possible to create a BIM of an existing building using a smartphone for 3D scanning.

Moreover, assuming this is possible, the objective will be to further investigate scanning by smartphone for BIM applications, by adding the following co-objectives:

- Combine smartphone 3D scanning with other scanning technologies.

- Establish a procedure for BIM creation from smartphone scanning.

- Determine the efficiency of the technology for construction purposes, by registering time spent for various steps of the Scan to BIM process.

- Measure geometric precision of the scanned model and of the final BIM.

- Discuss the usability of the technology and possible areas of application.

## 1.3   Approach

To achieve the objectives of this thesis, research and training in scanning and post processing is needed. Hence, the first period of work will be dedicated to exploring current technologies and research to establish a working smartphone Scan to BIM procedure. When such a procedure is established, an experiment which will test all the stated objectives is to be performed.

The experiment will include:

1. Locating a suitable building for the experiment.

2. Establishing control points in the building and measuring the physical distances between these.

3. Scanning the building using a smartphone.

4. Scanning parts of the building using a different scanning technology.

5. Processing the scan into a BIM.

6. Measuring the distance between control points in the scan and in the final BIM.

7. Registering the time spent on all parts of the experiment.

## 1.4 Limitations

The limitations of the project are the following:

- **Time**: The timeframe to complete the Master's Thesis is 20 weeks. This limits the complexity of the experiment and the time spent on establishing various Scan to BIM procedures.

- **Resources**: The equipment and software used for the experiment is limited to licences and scanning equipment which are free of charge or available for students at NTNU. The only purchase has been an Asus Zenfone AR, which was essential to acquire for completing the experiment.

- **Theory**: Chapter 2, *Technologies,* will elaborate on the theory and technologies laying the foundation for the experiment. The content of this chapter is limited to the basics of each technology. Readers who wants further insight in theoretical and technological matters are referred to the Bibliography.

# Chapter 2

# Technologies

## 2.1 3D Scanning Techniques

3D scanning is a technology used to create a digital 3D model from a physical object or environment. Scanning technologies can be dated back to the 1960s, where early scanners used cameras, lights and projectors [2]. Today, several scanning technologies are in use, each with its benefits and disadvantages. The shape and size of the object as well as the desirable level of detail are usually the decisive factors when a scanning method is chosen. In general, 3D scanning can be divided between contact and non-contact scanners. Non-contact scanning methods are most feasible for scanning large objects such as buildings, since they perform scanning without physically touching the object. This is usually done by emitting and detecting radiation or by regular photography. Figure 2.1 shows a typical way of performing 3D scanning today. The next subsections will focus on scanning techniques currently being used in the construction industry.

Figure 2.1: Ground fixed LIDAR scannig [1].

5

### 2.1.1   Laser Scanning

Laser scanners are part of a non-contact scanning group called *active scanners.* Active scanners emit radiation or light to sample surfaces. This is either done by detecting the reflection or analyzing radiation that passes through an object. Since a regular laser does not penetrate most surfaces, laser scanners obtain surface information by detecting reflections from the object. There are currently two main theoretical non-contact active scanning methods: *Laser triangulation* and *Time of flight.*

**Laser triangulation** uses trigonometry to determine the distance to a point on a surface. Figure 2.2 illustrates this principle. The figure shows a laser beam directed towards a surface. The reflected beam is recorded by a sensor placed within a short distance from the laser projector and the position is recorded. The sensor used is often a Charge-coupled device (CCD-sensor) which converts laser light into digital signals. The accuracy of this method is mainly determined by the angle between the laser projector and the sensor, hence it works best for short distances.
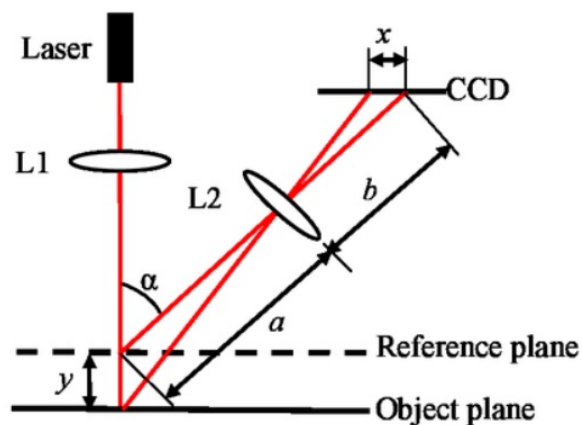


Figure 2.2:  Illustration of the triangulation method [1].

**Time of flight** is a method to determine a distance by measuring the time it takes for a pulse of energy to travel from an emitter, to a surface and back to the source. A laser beam consists of coherent light. Since the speed of light is known, the distance to an object can be determined by multiplying the speed of light with the time spent on the round trip. This number is then divided by a factor of two to find the distance from the emitter to the surface. The principle is displayed in Figure 2.3. 3D measurements to obtain global coordinates can be achieved if the laser emitters are equipped with rotary encoders [3]. When detecting pulses of energy that travels at the speed of light, the rate of detection is crucial for accuracy as the light beams travel extremly fast. Hence, accuracy of the Time of flight method is dependent on the distance from the emitter to the measured surface and the sensor's rate of detection. This method is thereby suitable for long distance surveying.
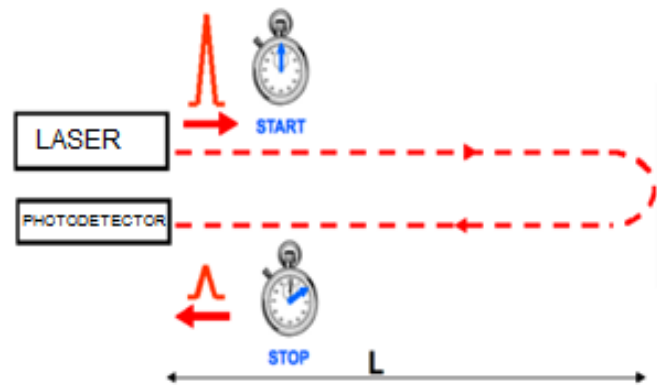
Figure 2.3: Demonstration of the principle of the Time of flight method [1].

A laser scan produces a raw *point cloud*. A point cloud consists of a vast amount of points, with each point containing specific data. The minimum amount of data each point need to contain is the 3D Cartesian coordinates of the point in a local coordinate system. Other information that can be included is the position of the scan in a global coordinate system and the colour of the scanned surface at the points position.

Today, the most applied technique for obtaining a point cloud using laser scanning is the Light Imaging Detection and Ranging method, also known as *LIDAR* [4]. For regular surveying it mainly uses the Time of Fight method but can also use *Phase shift measurement*, which is especially accurate for scanning moving objects or scanning while moving [5]. The technology is similar to RADAR. Modern LIDAR systems in the construction industry usually include additional equipment to make the scan products better:

- A positioning system, mostly using GPS, for determining the position in a global coordinate system.

- An Inertial Measurement Unit (IMU), to record orientation and velocity of the laser scanner [6].

- A photo camera to photograph areas scanned for easier post processing and better visualization result. This is often necessary for the point cloud to contain point colour data.

To obtain a fully covering 3D model and to bypass obstacles obscuring the target, scanning is done at multiple positions around the object. Using a ground fixed scanner, such as the LIDAR technology allows millimeter precision.
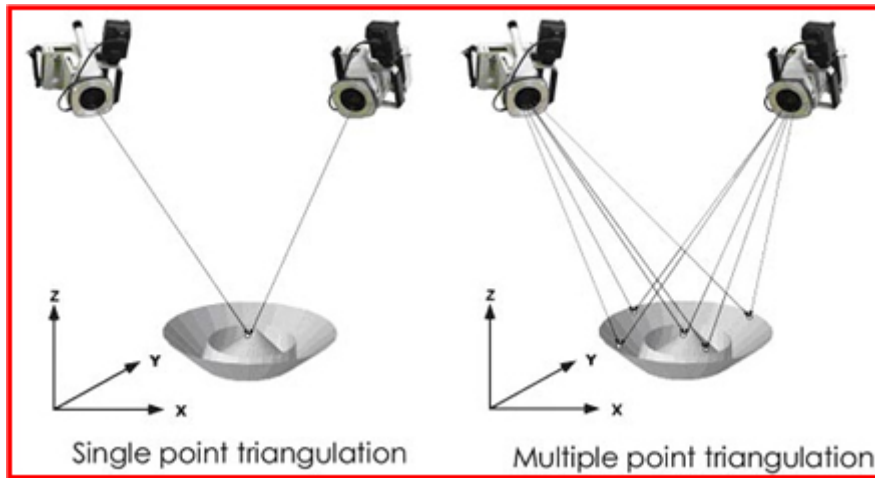
## 2.1.2   Photogrammetry



Figure 2.4: The principle of photogrammetry [1].

Photogrammetry is a non-contact passive scanning method, meaning that the scanners do not emit radiation themselves, but rather perform scanning by detecting ambient radiation. This ambient radiation is usually visible light but could also be from other sources, such as infrared light. Photogrammetry scanning is done by using a photo camera to detect visible light. The scanning is often conducted by taking photos of an object from several different angels. These photos are later post processed by extracting measurement from the collection of calibrated photographs and positioning them in a cartesian coordinate system. These techniques were developed right after photography was invented, around the 1850s. Thus, the techniques are well developed. Anyhow, applying photogrammetry for model creation has undergone a rapid development the last decades because of swift improvement in computer technology.

To make a 3D model out of photogrammetry, information on depth and distances in the photographs need to be acquired. This is done by locating the same points on different photos taken from various locations. Figure 2.4 shows how this can be done using either a single, or multiple points. The 3D position of these points are calculated by applying the trigonometric concepts of *parallax* and *triangulation*. For this to work, the exact location of the *origin* of each photograph is required. This position is determined via a process called *position resection* [7]. The distance between the two origins is then calculated as a displacement vector. The process of connecting all photos together in a coordinate system is typically done by optimizing a least-square function by the Lavenberg-Marquardt algorithm [8]. As the distance between the photograph's positions are known, virtual lines are drawn to similar points on the pictures, and thereby the distance is calculated by trigonometry. This concept is shown in Figure 2.5.
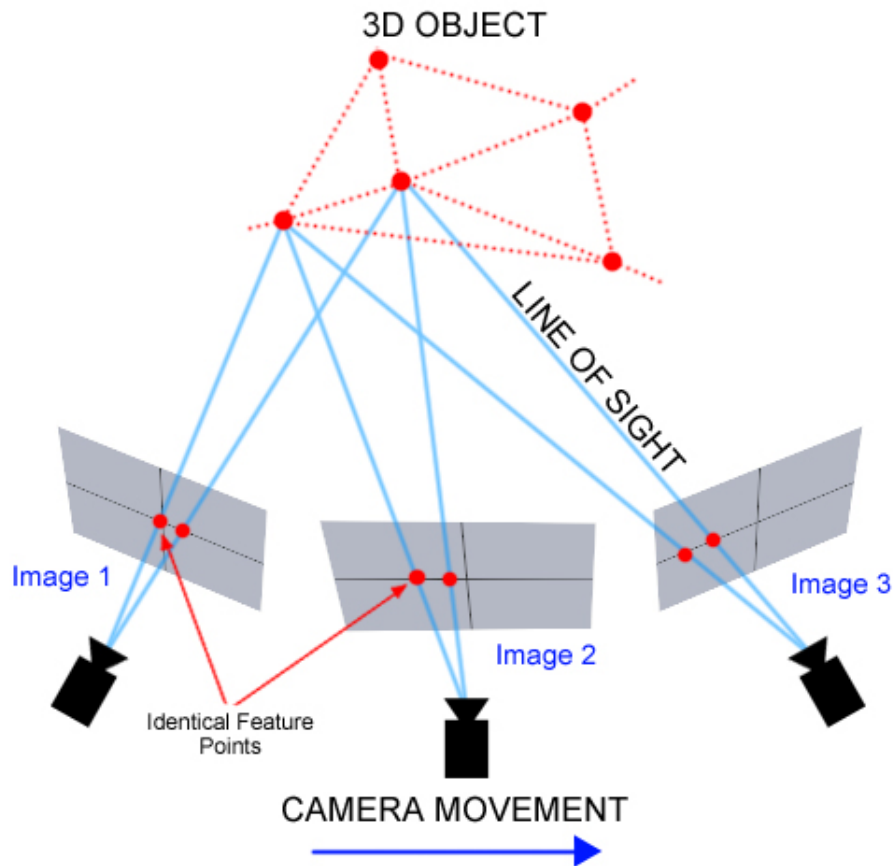
Figure 2.5: The concept of photogrammetry post processing for creating a 3D model [1].

Accuracy of the measurements is mostly dependent on three factors:

1. The pixel density of the photographs.

2. Physical distance from the photo camera to the object being modelled. The optimal angle between photos is 45 degrees.

3. The accuracy of the position of the origin.

Accuracy is typically about -/+ 5 cm, but could be decreased to 5 mm precision if necessary.

Photogrammetry is typically used for preliminary surveying of an area or a building. An Unmanned aerial vehicle (UAV) is often used to get an overview of the surveyed area from above. This allows fast and cheap scans of a vast area. UAVs are currently also being used for more detailed scans of the building's exterior. This has sometimes proven superior to laser scanning, as drones have access to the building from above and the scanning is faster. To obtain coordinates in a global reference system, geolocation of the origin can be done using a GPS. Photogrammetry has been used extensively for surveying purposes in construction projects for decades, but this experiment's application of photogrammetry is for buildings.

## 2.2   Scan to BIM

### 2.2.1   Builiding Information Model

Starting in the early 1970s, development of 3D models were based on early computer-aided design (CAD) efforts from several industries. Rapid development of computer technology provided the construction industry with several new tools both for calculations and design. Even though several industries developed 3D modelling to contain object-based parametric modelling throughout the 1980s and 90s, the construction industry stuck mostly to traditional drawings in 2D. In the early 2000s, BIM was introduced in some pilot construction projects. Since BIM allowed the content to include both physical and functional characteristics of a facility, it could be seen as a natural evolution of CAD for the construction industry [1][9].



Figure 2.6:  An illustration of components in a wall represented as a BIM object [10].

BIM software support the creation of objects. These objects consist of both geometrical and technical information. Figure 2.6 provides an example of how a BIM object could be structured. Dimensions and position are geometrical properties, while maintenance information, assembly and material properties are examples of technical properties [10]. Including all these properties to the model allows software to run various analysis of a building such as light analysis and fire simulations. Another main feature of BIM is usually the ability to model a building parametrically. Parametric modelling enables the relation between elements to be defined by certain parameters. These parameters are adjustable, which allows complex geometry to be manipulated more conveniently. An example is when the height of the columns in a building is adjusted, the position of floors and walls connected to the columns are also automatically adjusted. This combination of parametric modelling and use of objects is often referred to as *Object-based parametric modelling* [11]. Such an object can be seen in Figure 2.7.
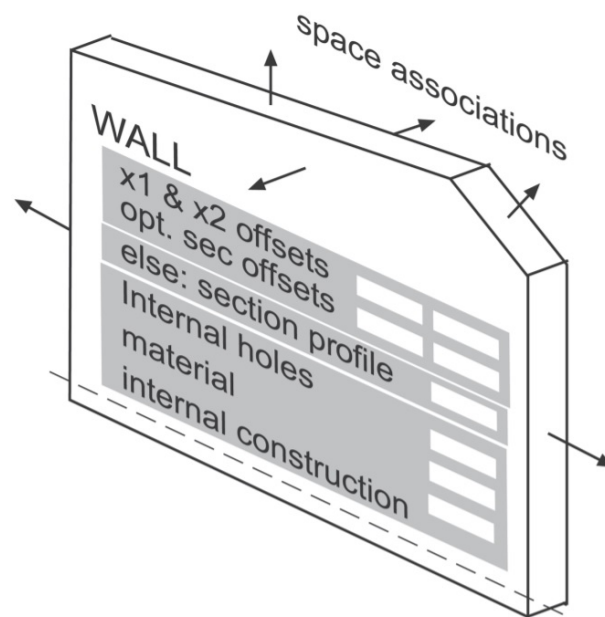
Figure 2.7: Illustration of object-based parametric modelling. A pre defined object with several adjustable parameters. In parametric modelling, these parameters include information about correlation to nearby objects [11].

Most BIM software also allows participants in a project to work and design simultaneously in the same model. This is helping the entire construction and designer team to visualize the project while it is being designed and built. According to several surveys, collaboration and visualization drastically reduces building time and construction errors. Thus, making it likely that BIM will continue to be a vital tool for construction projects in the near future [12].

### 2.2.2 BIM creation from 3D scanning

Most 3D scans taken for construction purposes today are never processed into a BIM. Often, a 3D model is created and used merely for visualization, detecting errors or as a basis for reconstruction. The process of generating a 3D model from laser scans or photogrammetry is currently a well-established process. On the contrary, generating a BIM out of scans is a relatively new and complicated process. Why is it this difficult to create a BIM out of scans? Generating a BIM means adding technical properties and relations between objects to the model. The geometry of the model can be extracted from a scan, but the technical properties, object classification and relation between objects are not easily extracted. These properties have to be recognized and understood, making it easier to create the BIM manually [1].

Despite difficulties, there are some promising algorithms for automated object recognition and BIM generation. The most promising algorithms can be separated according to approach into two main categories: *Global optimization approaches* and *Local heuristics*.

**The Global optimization** approaches enhances the idea of recognizing big object groups and dividing these into smaller parts. Recognition is often based on statistical methods, where tools such as a *Bayesian model* or a *Markov Chain Monte Carlo algorithm* is used to find parameters that maximize the possibility of recognizing building elements [13][14]. An example of such an approach could be for an urban model. The algorithm would first segment the model into parts like *Building, Vegetation, Ground and Clutter.* The *Building* segment may later be segmented into: *Walls, Floors, Beams and Columns* and so on [15].

**The Local heuristic** approaches assume that a geometric model can be decomposed into simple parametric surfaces. The algorithms are based on detecting orthogonality or symmetry to recognize objects and their relation to each other. A flat horizontal surface leading to an orthogonal flat vertical surface could be expected to be a floor leading to a wall. If there is a comparable situation in the next room, the gap between the two vertical surfaces could give an indication of the thickness of the wall as well. Algorithms like this are based on assumptions that walls are vertical and floors are horizontal, hence they are prone to error, and many objects will not be categorized [15].



(a)            (b)            (c)
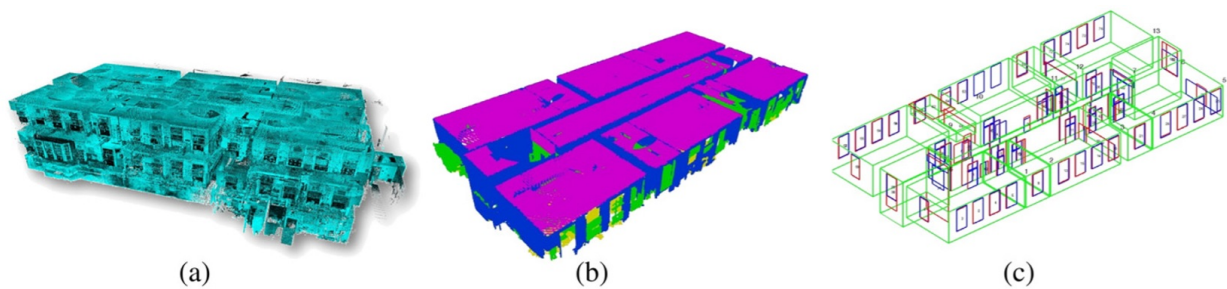
Figure 2.8: Exampel of an algorithm for automatic object recognition:
(a): The scanned point cloud.
(b): The algorithm recognizes and models key structural components - walls (blue), floors (yellow), ceilings (magenta, and clutter (green).
(c): Detailed surface modelling detects and create objects like windows, walls and doors [16].

An example of an automated object recognition algorithm is displayed in Figure 2.8. There are currently some software for automated or semi automated Scan to BIM under development based on these algorithms, but no remarkable success has been made so far. Meaning that the BIM generation process is mainly manual, time consuming and error prone.

### 2.2.3  Areas of application

As seen in Figure 2.9, BIM is currently being used for a vast amount of applications within the construction industry. Scan to BIM can't be used in all these categories but has the potential of being used regularly in a few of them. In application areas like renovation and demolition, Scan to BIM is being used to some extent today and there are clear indications that it has positive impact on the projects. In other areas such as documentation, analysis, operation and maintenance, the technology also have some potential to provide better and more effective projects. This is making it likely that there will be even more usage of Scan to BIM in the future.
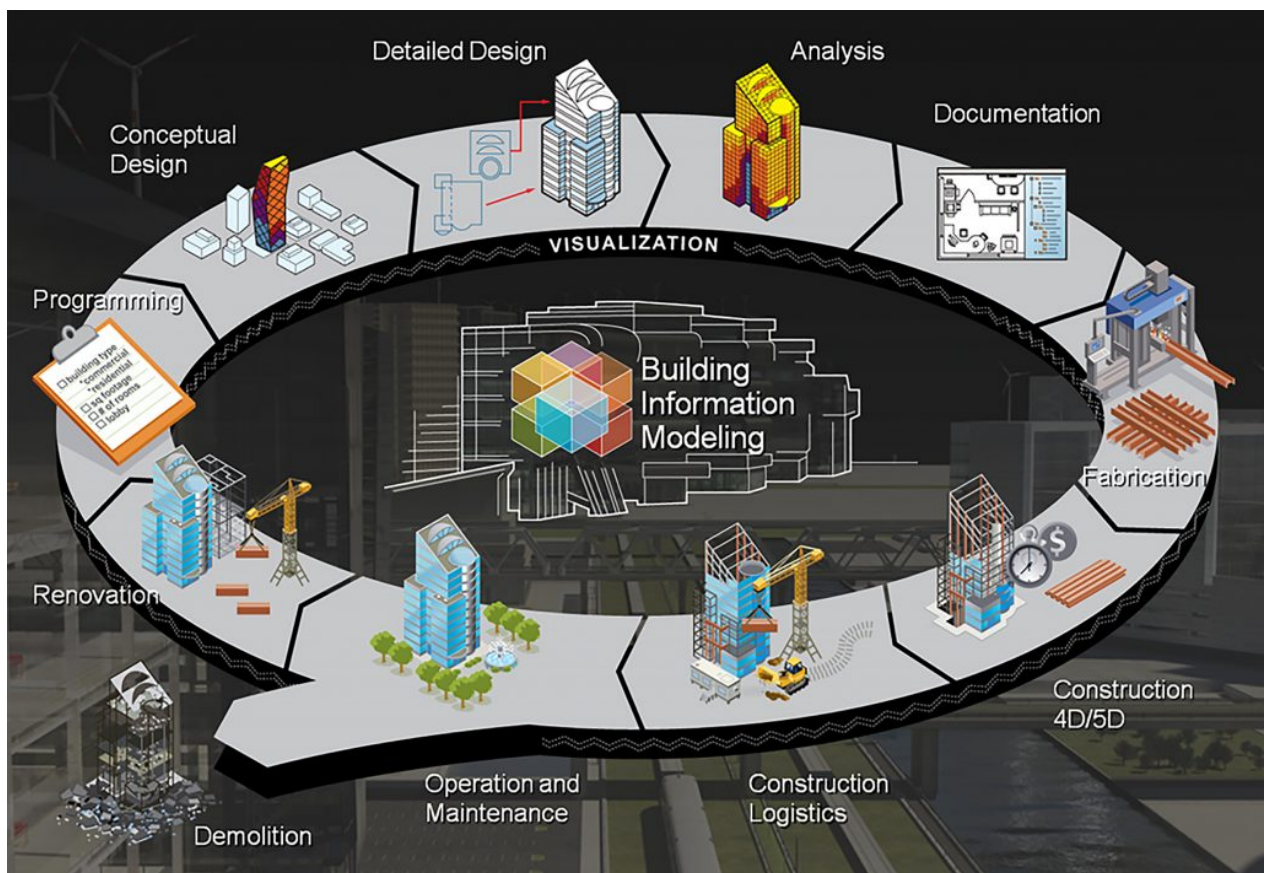


Figure 2.9: BIM: Areas of application [17].

Utilization of Scan to BIM has so far been most prominent in these four areas: [1]

1. As-Built documentation is a document handed over to the project owner after a construction project is finished. These documents show how the building was built. As changes often occur during construction, the As-Built documents often deviate from the original drawings. Scanning a finished building and comparing the As-built model to the original model provides a good foundation for comparison instead of using manual measurements and drawings.

2. Scanning as a basis for modelling and structural analysis are often done in rehabilitation and rebuilding projects. This is done to acquire the exact geometry or the entire building in a construction site in an effective way. The scan is either used for further modelling and BIM creation, or as a foundation for the structural analysis as dimensions can easily be obtained from the scan. This tool is especially convenient to acquire accurate geometry when the original drawings are missing or non-existing.

3. Scanning for visualization and exploration of rebuilding options are often done by building owners who either want to know how their buildings can be better utilized or to obtain a model for visualization. When the owner has a model of the building, it is possible to make changes and adjustments and see how it will influence the building properties. A model for visualization may for example be used for marketing the building.

4. Revealing errors during construction, especially in the interface between two construction disciplines, is often difficult and mistakes are often discovered too late. Small errors, for example a minor misplacement of a recess, can cause big delay and extra costs for the contractor. By continuously scanning areas during construction and checking the result against the design drawings, errors can be detected before they cause a delay.

## 2.3   Post Processing Software

The need for post processing is based on the intended application of the scan. A scan is processed because the *Raw scan* usually do not fulfill the application purposes. There are many ways of processing a scan, but some of the most common ways of processing are by: Aligning and merging several scans, *cleaning the scan* using various software tools, smoothing or parametrizing the scan, generating a mesh and creating scan regions [18] [19].

A 3D scan contains either a point cloud or several pictures depending on the scanning technique. Laser scanners produces point clouds. The laser scanner manufacturers tend to have their own software packages for their equipment. It is often specialized to handle the specific file format that the scanner exports. The software can be used for diverse variation of post processing and can even be linked with CAD products like AutoCAD or Revit.

If the scan consists of pictures, it needs to be processed using photogrammetry into a 3D model. Photogrammetry post processing software is often produced by companies which specialize specifically in this discipline. Even though the technique of merging the photos into a 3D model is similar, the software uses various algorithms to smooth and clean a model after it is created. Some popular software packages for this includes: *Pix4Dmapper Pro, Agisoft PhotoScan professional* and *Bentley ContextCapture*. The post processing software often

allows creation of a *polygon mesh* or a point cloud. A polygon mesh is a computational way to describe a geometrical surface by dividing it into small components, usually triangles, quadrilaterals or convex polygons. These shapes are collections of vertices, edges and faces. They contain data about the surface of the object gathered from the scan. This is described in the text of Figure 2.10. Figure 2.11 displays how the number of vertices influence the mesh quality [1][20][21].
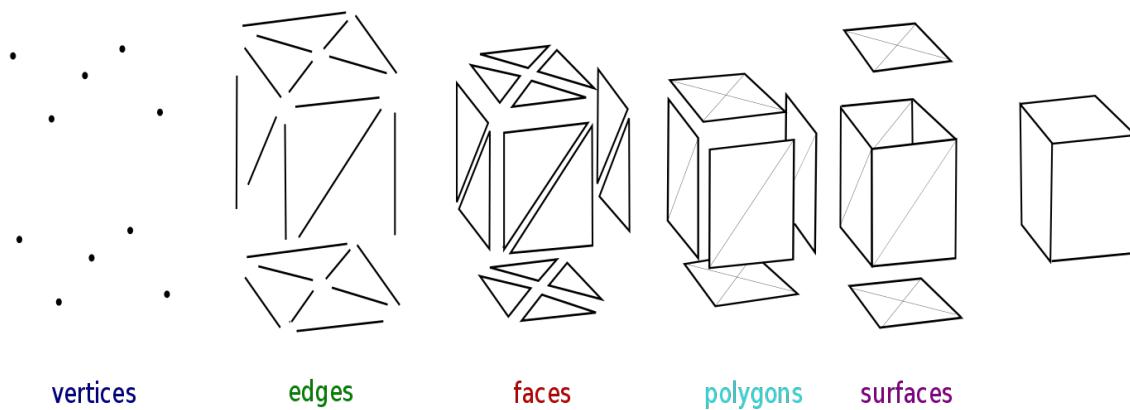


vertices      edges      faces      polygons      surfaces

Figure 2.10: The components of a polygon mesh: **Vertices** often include position as well as information such as color, normal vector and texture coordinates. **An edge** contains data about the connection between two vertices. **A face** is a closed set of edges that can include information about the encircled area [20].



(a) 25,000 vertices.   (b) 5,000 vertices.   (c) 500 vertices.
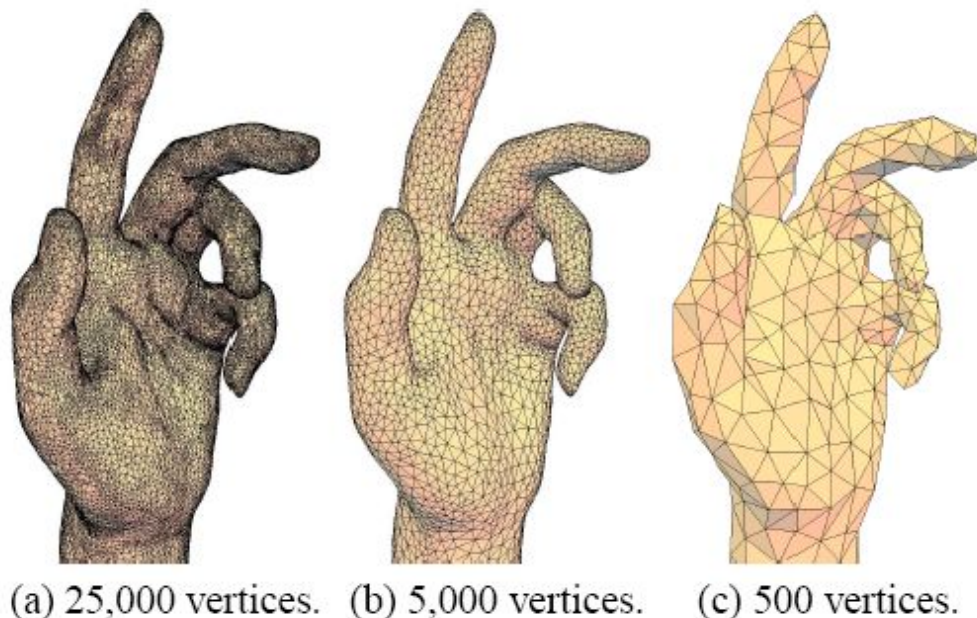
Figure 2.11: Increasing the number of vertices gives a better approximation of the surface [22].

The following sub sections will further elaborate on specifics about the software packages

used in the experiment of this report:

### 2.3.1   Mesh and point cloud processing software

Some software packages are developed merely for handling and processing 3D models which are already made. Two of the most applied software packages for this are *MeshLab* and *Cloud Compare*:

**MeshLab** is an open sourced system for processing and editing triangular meshes. The software has been developed by the ISTI – CNR research centre in Italy since 2005 [23].  It is used for handling everything from scanned data to models which are ready for 3D printing. The program is available for Windows, Linux and Mac OS X and support the most common file formats both for importing and exporting files. Supported file formats include: *Ply, stl, obj, off, 3ds, vrlm 2.0, u3d, x3d* and *dae*. Information about the file formats can be found in Appendix A. The software is most feasible for processing single objects, like a person or an artefact, but can also be used on larger areas. There are several different areas of application including preserving cultural heritages, paleontology, microbiology, surface reconstruction and rapid prototyping [24].  An example of processing in MeshLab is displayed in Figure 2.12.

The key features for processing in MeshLab are: [24]

- **3D acquisition: Aligning, reconstruction and colour mapping:** This include tools for moving and transforming separate meshes into a common reference system.  This is also referred to as *mesh registration*.  It also includes algorithms for merging several meshes or point clouds into a single mesh, reconstruction of missing parts in the meshes as well as tools for projecting colour to the model. This is often done either by automated algorithms or manually with assistance from the software.

- **Cleaning 3D models:** Scans often produce some surface errors or duplications. These vertices are often undesirable in the 3D model; hence the user would want to remove them.  MeshLab offers a series of automatic, interactive and semi-manual filters to remove these superfluous vertices.

- **Scaling, simplification, refinement and remeshing:** By applying scaling the user can either fit different scans to the same mesh or rescaling objects to fit other purposes. Meshes could also need to be adjusted, either by remeshing, simplification or refine certain parts of the mesh for a specific application.

- **Measurement and Analysis:** The software supports the possibility to extract various geometric information like curvature, point to point and geodesic distances or local vertex density.

- **Visualization, presentation and 3D printing:** MeshLab can export data in various formats for visualization, 3D modeling and 3D printers.



Figure 2.12: An example of MeshLab processing: Smoothing [24].

**Cloud Compare** is a 3D model processing software that specializes in point cloud processing, but can also handle meshes. The software treats almost all 3D entities as point clouds. A triangular mesh is typically treated as a point cloud with associated topology and information. Cloud Compare is available for Windows, Linux and Mac OS X platforms, and supports more file formats both for import and export than MeshLab. Supported file formats include: *bin, las, E57, ply, obj, vtk, stl, off, fbx, dxf, pov* and *icm*. This allows Cloud Compare to handle more specialized point clouds file formats than MeshLab. Another different characteristic of Cloud Compare compared to MeshLab is that it focuses more on scanned areas and complex structures while MeshLab focuses on objects [25]. An example of processing in Cloud Compare is displayed in Figure 2.13.
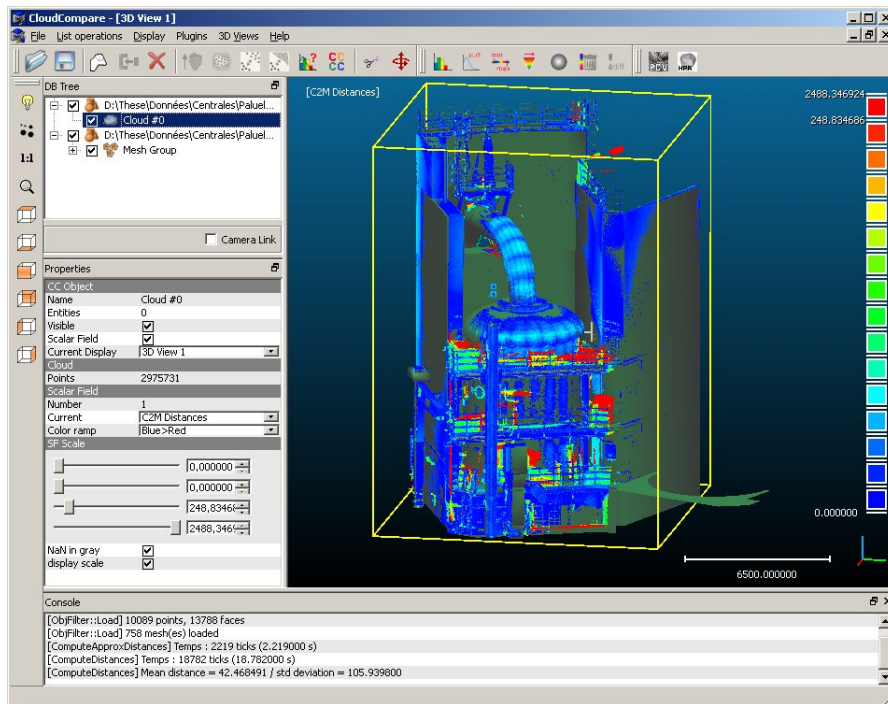
Figure 2.13: The interface of Cloud Compare while processing a structure [25].

The software was originally a tool designated to perform direct comparison between dense 3D point clouds but later evolved towards a more general 3D data processing software. Today, the software is open sourced with a lot of the same key features as MeshLab. Differences include that Cloud Compare relies on a specific octree structure that enables exceptional performance for handling certain point clouds. It also has some more advanced processing algorithms like local statistical filtering, specialized RANSAC algorithms and more [25].

### 2.3.2 BIM preparation software

Presently, there are no software that automatically generates a BIM from a scan. However, there are some software with semi-automated object generation and other features that makes the BIM creation easier. Autodesk Recap is a software with such features, and it therefore helps preparing the point cloud to make further processing in Revit easier:

**Autodesk Recap** stands for *Autodesk Reality Capture* and is a software developed by Autodesk to process scanned data into manageable models ready for further design. Since point clouds tend to contain a huge amount of data, which are often hard to manipulate and edit, there has been a need for a software to make the point clouds more workable. The software is mainly focused on laser scan point clouds but have some options for photogrammetry modelling as well. An extension to the software is called Autodesk Recap Photo, which allows the user to upload a series of photos of an object or environment. These photos are automatically processed by the software to generate a 3D model. An example of processing in Recap is shown in Figure 2.14.
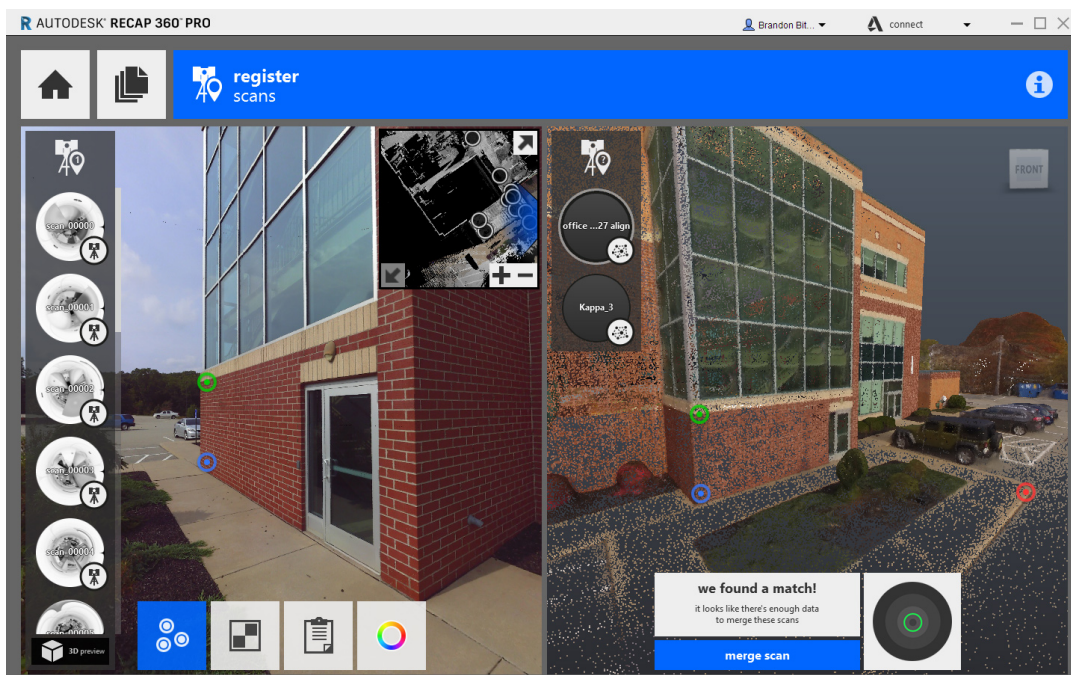


Figure 2.14: Autodesk Recap processing example [26].

Recap imports data from wide range of file formats, such as: *Rcs, pts, las, E57, xyz, rcp* and *prj*. Exports are limited to the file formats: *Rcp, Rcs, pts* and *E57*. Both *rcp* and *rcs* are easily exported to Revit, hence Recap is an appropriate tool to prepare a point cloud for further processing in Revit.

Useful features of Autodesk Recap include: [26]

- Tool for merging point clouds by adjusting the origins of each cloud.

- Annotation and colour adjustment.

- Creation of regions by selecting groups of points. The user can choose to select all points within a scaled box, or to choose all point on a specified plane. This allows the user to manually divide the point cloud into floors, walls and object. These regions can later be exploited in Revit modelling.

Another software worth elaborating briefly, even though it was not used during this experiment, is Edgewise:

**EdgeWise** is a software developed by ClearEdge3D focusing on BIM and automated BIM creation. It is divided into four main software packages: *Structure Modelling, Pipe Modelling, Duct Modelling* and *Building Modelling. Structure Modelling* is shown in Figure 2.15. Each package has some features to make modelling out of scanned data easier. For example, Edgewise Structure Modeling can be used to search laser scan data for structural elements like steelwork, pipework, beams etc. and then insert a solid model object in that position. This is a semi manual process, where you draw a fence around the object you want to model, and the software inserts the correct size of the appropriate object, one piece at a time. This is often time saving compared to manual modelling. Another example is Edgewise Building Modeling which uses advanced algorithms to automatically identify walls and windows in the point cloud and export these as Revit family objects [27][28].
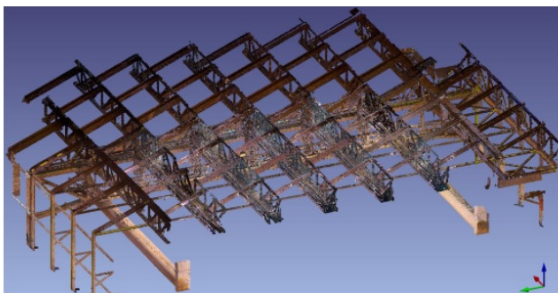


Scan Data in Edgewsie
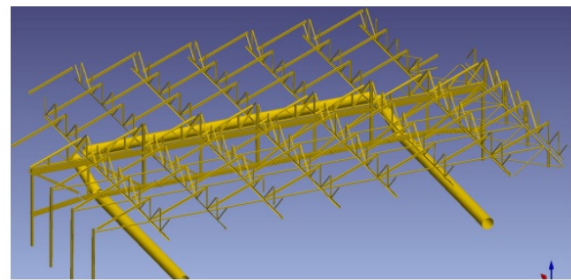
Steel Model created from the Scan data

Figure 2.15: *Edgewise Structure Modelling* used to extract the structure of a building [28].

### 2.3.3 BIM software

BIM usage and software have proliferated during the past decades. Autodesk Revit is by far the most used today [29]. The software has suitable applications for architects, landscape architects, MEP-engineers, designers, contractors and structural engineers. The layout of Revit is shown in Figure 2.16. Revit is also an Autodesk product, well suited for interconnection with Autodesk ReCap. This makes it an excellent choice as the BIM software for this experiment:

**Revit's** first version was released in April 2000. This makes Revit one of the first dedicated BIM software packages [30][31].
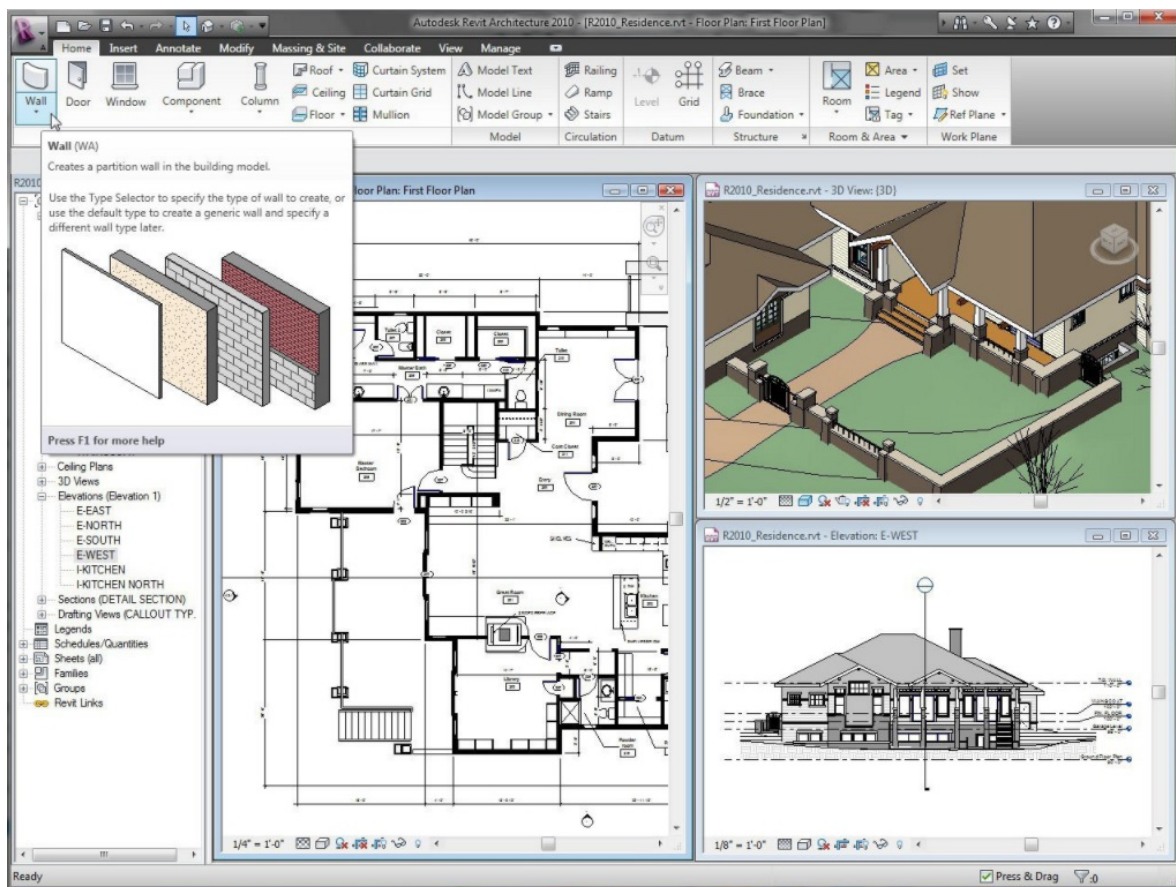


Figure 2.16: The Revit interface, using both 2D and 3D view at the same time for easier modelling [32].

Designing in Revit is based on the creation, manipulation and placement of objects. These objects are referred to as *Families* in the Revit terminology. There are three main categories of families used in the software: [30]

- **System families:** Already built in the software, with the option of being reshaped, changed or created by the user as well. This group include walls, floors, roofs, ceilings and more.

- **Loadable families:** Are created as separate projects and saved as a family. These objects are later loaded into a project to be used. This gives users the option of making a database of premade BIM objects. There are currently several web pages on the internet which are used as databases for objects like this. Most of these sites are free of charge and allows designers to download premade families [10][33]. This can include objects like doors, sinks and extrusions.

- **In-situ families:** Are families created while working on a project. This objects can be copied and reused in the project or exported as a loadable family. They also have the same toolset as loadable *families*.

All Revit families are parametrically made, meaning that it is possible to change all parameters of the objects, for instance length and width. If you change the family, all objects placed in the project, as this family, changes.

Other key features of Revit include:

- The possibility of user collaboration in the same project, meaning that users can work, design and watch the same model at the same time.

- Tools for creating topology and surfaces.

- Tools for simulating weather and sun conditions. This is either done by manually creating a weather situation or based on GPS coordinates of a specific location.

- Separates between classes of objects like structural, architectural and MEP.

- Allows several types of analysis like structural analysis, energy analysis and more.

- Importation of point clouds, images, new materials, objects, other 3D models and CAD products.

- Collaboration with other software packages like Dynamo Studio or Autodesk Robot Structural Analysis [34][35].

## 2.4 Google Tango

### 2.4.1 Background

Project Tango was initially released by the *Advanced Technology and Projects* division of Google in 2014. Tango is an Augmented Reality (AR) computing platform, with the purpose of allowing mobile devices to detect their position and surroundings merely using computer vision with no external signals like GPS or Bluetooth. During 2015, two devices were made and put for sale to demonstrate the Tango-technology. In 2016, Lenovo released the first cell phone with the Google Tango platform while Asus released another one in 2017. There are currently only two commercial phones with the Google Tango platform, but other companies have developed specific devices that exploits the Tango technology. An example of this is the Canvas structure sensor, developed specifically for scanning buildings [36][37][38].

The project was launched on an open system basis, allowing app developers all around the world to create apps for the platform. Thereby, Google allowed the users to develop new areas of application for the technology. By now, Tango is currently being used for a broad range of purposes, examples include: [39]

- **Tango for games:** Utilizes the device's ability to connect reality to the phone to make new and more realistic games.

- **Tango for shopping:** Gives the user the ability to check whether cloths will fit the user or how various kinds of furniture will fit in the house.

- **Tango as a measuring tool:** Scanning an object or an area gives the user the ability to conduct measurements in the scanned area. This can be used to calculate the size of an area or as a database of various measurements.

- **Tango for navigation:** Since Tango utilizes computer vision to navigate, it can be used for navigation or mapping in areas without GPS reception. This is particularly useful for navigation or for tracking your movements inside buildings where there is no available GPS signal.

- **Tango for holograms:** As it is possible to scan a person and later display this person in the room, Tango can be used to create and manipulate holograms. An area of application might be for business meetings or other gatherings where a person is unable to join.

In late 2017, Google announced that they would shut down the Tango project in favour of the new *ARCore project*. This would be effective from March 1st, 2018. The reason for the discontinuation was the increased competition in the AR market, as the competitors focuses their AR projects on phones without the specific hardware components needed on a Tango device. The goal of ARCore is the same as for Tango, but Google is now trying to use regular

phones instead of phones with Tango hardware. Without the Tango hardware, the accuracy and efficiency of the scanning is likely to drop, but it also means that it will make the technology available for all regular cell phones. Google is transferring most of the Tango technology to the ARCore project. Since ARCore uses the same technological basis as Tango, the technology used in this thesis will still be relevant for further work, even as project Tango is shut down [40][41].

### 2.4.2   Hardware and fundamental concepts

A Tango device contains hardware that makes it able to navigate, scan and recognize areas. The base equipment used for this are motion tracking cameras and an infrared 3D depth sensor. There are usually two or more cameras installed to provide "human eye attributes", hence depth perception. The IR depth sensor allows the device to accurately measure distances utilizing the Time of flight method. Tango devices often include other sensors like accelerometer, gyroscope, ambient light, barometer, compass and a GPS [42].

Google Tango has embedded three main fundamental concepts to achieve reality capture:

**Motion tracking** is done so the device can constantly record its position relative to its surroundings. Tango does this by combining the inputs from the device's cameras and an IMU. The process is called *Visual-Inertial Odometry* [43]. The device's cameras are used to detect casually distinct features in surrounding areas. These points are called feature points and are used to compute the device's change in location. The IMU consists of accelerometers and gyroscopes, used to track the body's specific force and angular rate. This gives the device the possibility to calculate how fast it accelerates and what direction it is turning. An illustration of this principle is shown in Figure 2.17. Combining these two features gives the device the possibility to estimate its position and orientation of the camera relative to its surroundings over time. In order to keep motion tracking even when moving fast, the camera is able to capture and analyse up to 60 images/second. All tracking is based on a local 3D coordinate system created from the second the tracking starts. The origin is the starting position of the device and all other positions will be recorded with reference to the origin [44][42].
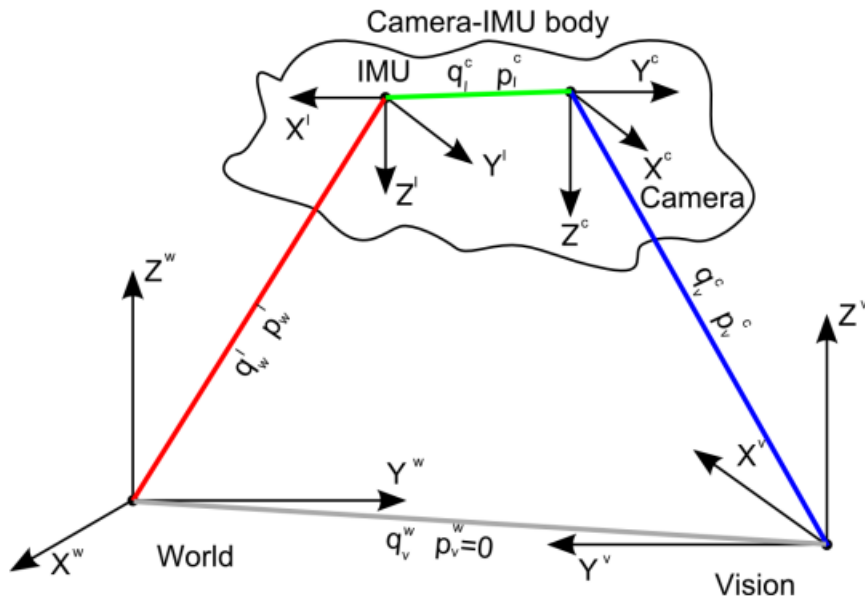
Figure 2.17: The coordinate systems of the IMU and the Camera need to be connected to the representation of surrounding surfaces [45].

**Area Learning** extends the motion tracking data, so the device is able to remember areas. This is done based on the Simultaneous Localization and Mapping (SLAM) approach [46]. It includes storing features from recognizable points captured by the camera during motion tracking. These are applied to recognize environments. The files are called Area Description Files (ADF). Area learning is used to improve the accuracy of motion tracking by storing the area while tracking. An example of an area learning algorithm is shown in Figure 2.18. Motion tracking alone is bound to error as a small error in either the IMU or the camera's feature detection will propagate while moving around. Area learning is also used for several other Tango applications [42][47].
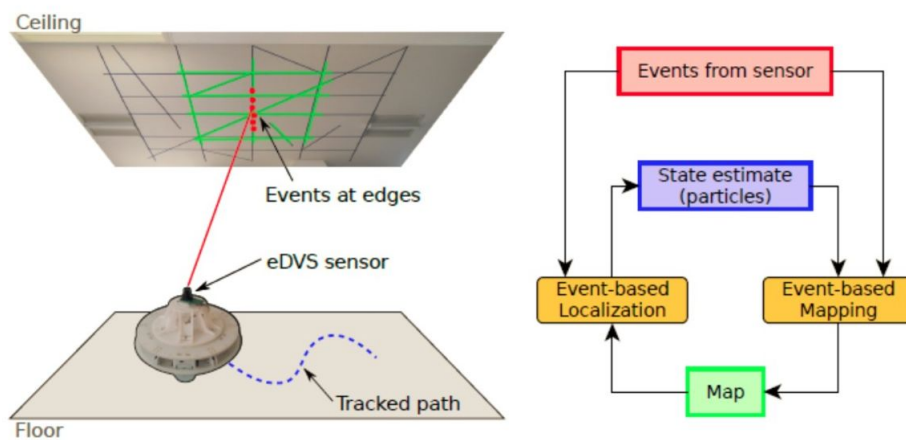


Figure 2.18: An example of a basic area learning algorithm combined with a sensor and the tracked path [48].

**Depth perception** is achieved in Tango by combining three 3D scanning techniques: Structured light, time of flight and stereopsis [1][49]. The structured light and time of flight method utilizes the IR depth sensor of the device and can measure distances quite accurately. Stereopsis uses the phone's cameras which are located slightly apart to gain the depth perception. When performing a scan or measuring depth, the device relies mainly on the IR depth sensor and the time of flight method, because it is the most accurate. The device still has the option of relying on for example stereopsis when light conditions will not allow the IR depth sensor to measure accurately. When creating a 3D model using Tango apps, the depth perception is often combined with the phone's cameras to gain a coloured model. An example of how this can be done is shown in Figure 2.19. All raw 3D scanning exports from Tango devices are generated as point clouds, but some apps are converting the point clouds into meshes.
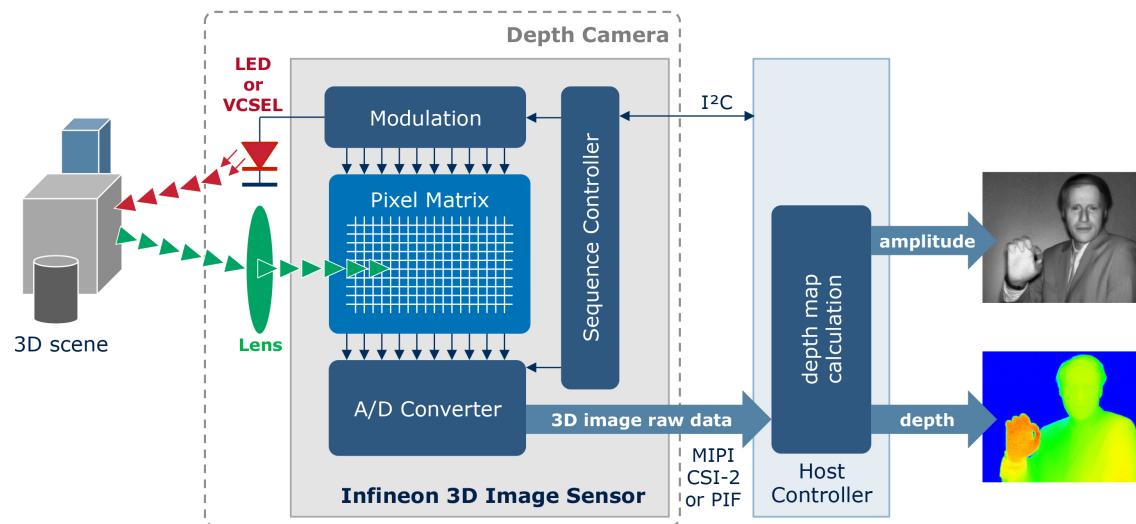


Figure 2.19: An example of a 3D modelling algorithm based on stereopsis depth perception [50].

By combining these three key features of Google Tango, a device can move and scan areas quite accurately, making it feasible for Scan to BIM.

### 2.4.3 Tango scanning apps

Since the Tango system is open-sourced, various applications have been developed for 3D scanning. Some of these apps focus on object 3D scanning while other have their focus on area or building scanning. These apps use a combination of the Tango futures together with scanning algorithms to gain a mesh or a point cloud of the phone's surroundings. This is done simultaneously while scanning, making it is possible to see the 3D model evolve on the screen while you are scanning. Even though many of these scanning apps produce the same result, there are some minor differences. Key areas where it is possible to see some difference is in the resolution, speed and scanning range. A higher resolution will provide a more detailed scan, but can at the same time cause big files which are hard to run. The scanning range and speed is really determined by what kind of object or areas the scan is intended to. For detailed scanning of small objects, there is no need for a huge scanning range, while in area scanning it could be necessary. As a part of project Tango, Google released an app for scanning buildings called *Constructor Developer Tool*. This app has resolution, speed and scanning range options suitable for scanning rooms inside a building. The app will be refered to as *The Constructor app* for the rest of this report. Other companies have developed similar tools like Matteport with *Matterport scenes* and Imerso with the *Imerso app* [51][52][53]. A screenshot from the Matterport scene app is shown in Figure 2.20



Figure 2.20: A screenshot from the Matterport scene app with added dimensions [52].

# Chapter 3

# Project Workflow

## 3.1 Training and Preparation

Due to lack of literature on smartphone scanning for BIM creation and lack of experience in 3D scanning, there was a need for preparations before the experiment. This section will elaborate on the most important steps that were conducted for learning and preparation. It was done to understand how to perform a scan and further process the scan into a BIM. An overview of the file format transformation is shown in Figure 3.1.

### 3.1.1 Scanning and processing software

After several days learning how to preform a scan and convert it to a 3D model feasible for BIM creation, the result was the following solution:

**Scanning**

Using the pre-installed constructor app on the Asus Zenfone AR, provided an uncomplicated way of scanning individual rooms and merging several rooms together in a single scan.

**Post Processing**

1. The constructor app was capable of post processing the raw scan on the smartphone. The data file could further be uploaded to Google drive, making it accessible online.

2. The constructor app exports the scans as either *ply* or *obj* file formats. These are not compliant with neither Autodesk Revit nor Autodesk ReCap.

3. A workaround is to use MeshLab as a transformation link between the constructor app's export and Autodesk ReCap. A major disadvantage was loss of colour data, since the *xyz* format exported from MeshLab carries point coordinates only. The BIM creation would benefit from coloured points.

4. The MeshLab alternative Cloud Compare provides a way of converting a mesh into a point cloud without loss of colour. Moreover, Cloud Compare even exports E57 file format, which could be imported by Autodesk ReCap and further to Revit.

5. Processing from Asus Zenfone AR via Cloud Compare allows editing and point cloud preparations both in Cloud Compare as well as in ReCap while keeping the coloured point cloud file format, making it a feasible way of doing the post processing.



Figure 3.1: The software workflow and file formats.

### 3.1.2  Tests

To further understand in detail how to perform the experiment, several tests were conducted for both scanning and processing. In this section the main tests are described:

**Test 1: Scanning software**

The phone comes with the Constructor App preinstalled. There are, however, other scanning apps like Materport Scenes and Imerso that could have been used as well [52][51]. Test 1 was a comparison between the Imerso and the constructor app. Table 3.1 explains the experienced differences between the software packages:

Table 3.1: Test 1, comparison between Imerso and Constructor.

| Attribute | Constructor | Imerso |
|---|---|---|
| **Scanning** | Constructor seems to have better visual quality on the screen while scanning. | Imerso can embrace more of the room at once while scanning, hence making the scan faster. After scanning for a while, Imerso lag more than the Constructor app. |
| **Time** | Constructor was slower in general. | The Imerso scanning was generally faster until it started lagging. |
| **Errors** | Not a big difference between the apps. Both apps *lost tracking* and crashed several times. | Imerso crashed some times during processing. |
| **Quality** | Not a big difference, but the scan is more coarse than Imerso. | Not a big difference, but Imerso's scans looks smoother and has less holes than Constructors. |
| **Processing** | The constructor app doesn't have any options regarding processing, it is simple and produces only *ply* or *obj* files. | The Imerso app allows the user some different processing options and you can also choose your export format from a wide range of file formats. The smartphone processing takes more time and crash more often than Constructor's. |
| **File size in the main test** (the same area) | Constructor: A *ply* file at 58 MB, while E57 file with 3 million points at 79 MB. | Imerso: A *ply* file at 10 5MB, while E57 with 3 million points at 43 MB. |
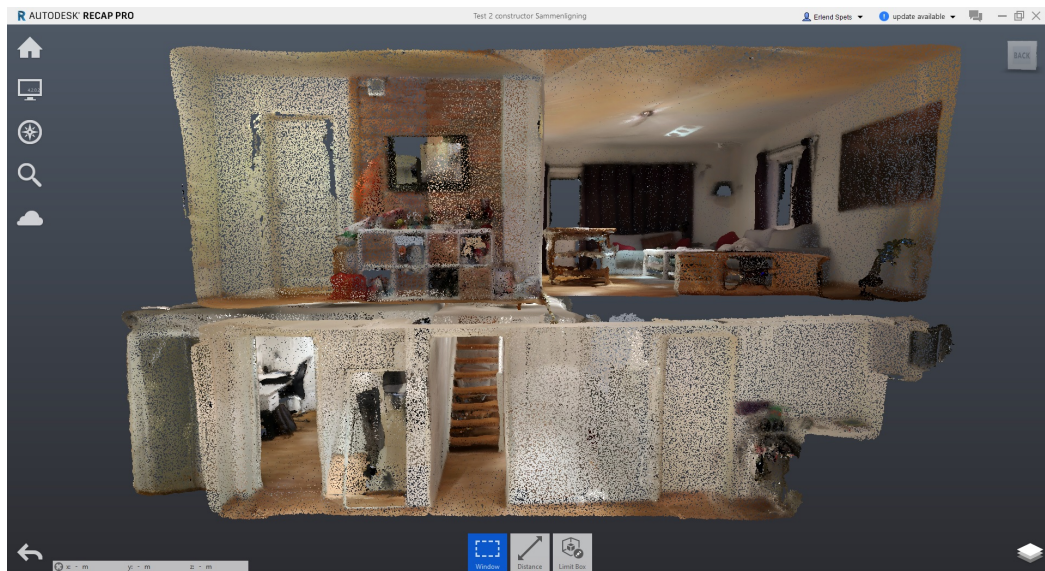


Figure 3.2: Test scan using The Imerso app.
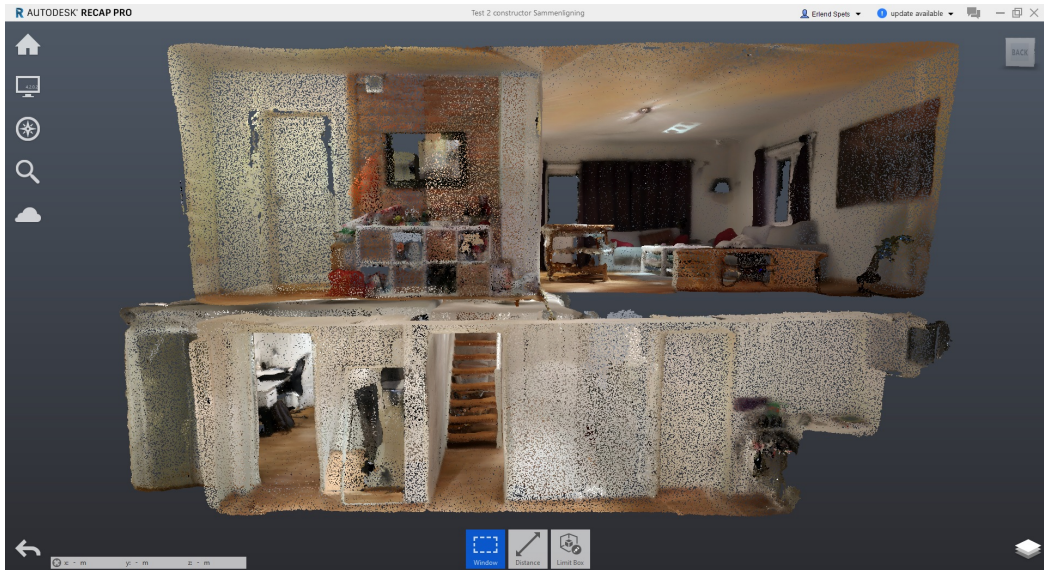
Figure 3.3: Test scan using The Constructor app.

As Figure 3.2 and 3.3 show, the scanning quality is similar. As both apps have their benefits and disadvantages, the apps are quite equal in quality. The constructor app is more accessible as it is pre-installed and free, making it easier to use for construction companies that wish to utilize this technology. Thus the Constructor app will be used for the experiment.

**Test 2: Light conditions**

Scanning by using the phone's cameras is influenced by the ambient light conditions. Various situations were tested to figure out the light influence. Figure 3.4, 3.5 and 3.6 show the result of scanning at night with the room's lights on, in daylight with room's lights on and in daylight with the room's lights off:
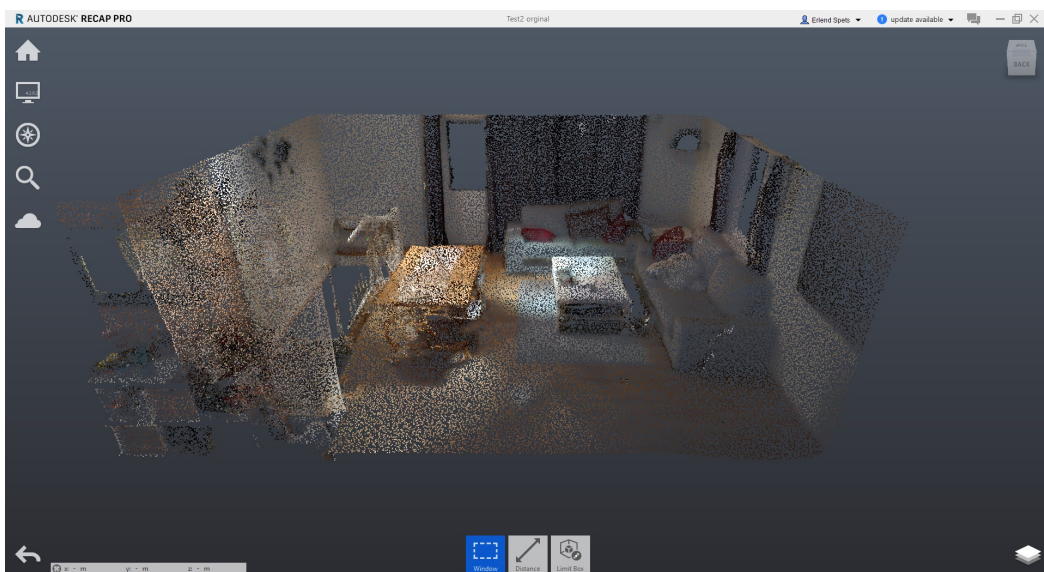


Figure 3.4: Scanned room during night.

Figure 3.5: Scanned room during daytime with artificial lights on.
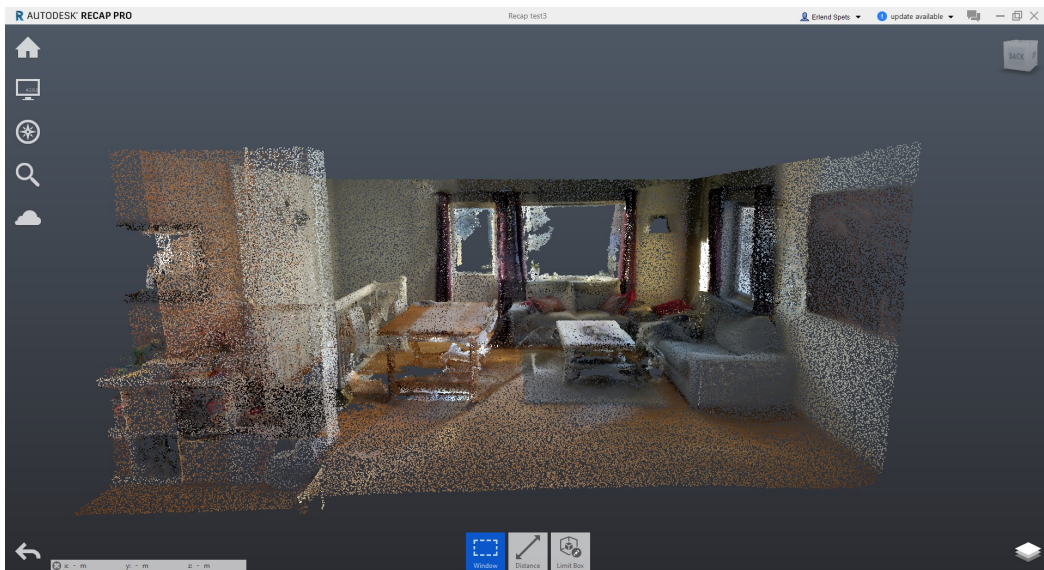


Figure 3.6: Scanned room during daylight with artificial lights off.

The result shows that there are no major differences between the light conditions. Some lessons learned from this test:

- Daylight enhances detail.

- Windows are scanned more smoothly in the night when no sunlight is hitting the camearas.

- During daytime, there is no significant differences between having the lights on and off.

- If parts of the scanned area are to dark, scanning stops, with the error message: *To*

> *dark! Use the device in well-lit areas.* This is usually not a problem and the scanning presume as you turn the phone to an area with more light but can sometimes result in the scanner losing tracking. The app will then add the newly scanned areas with a slight error to the previously scanned areas, making the scan useless.

The general conclusion is that scanning proceeds well in most light conditions except too dark. Scanning during daytime or in well-lit rooms will provide the highest scanning quality.

**Test 3: Measuring distance**

As a part of the experiment, distances between fixed points in the building were measured in order to compare with the same distances in the scanned 3D model. To find an appropriate solution to this, various ways of making the control points and ways of measuring distances were tested. Most of the points were made by drawing various kinds of markings on sheets of paper and fastening these on surfaces in the room. The equipment used for the test are shown in Figure 3.7. The other points were picked as edges of various objects within the room. After doing that, the distance between these points were measured using a yardstick. After finishing this, the entire room was scanned and the product was opened both in Cloud Compare and MeshLab. As Cloud Compare is based on viewing all meshes as point clouds, MeshLab provides a more feasible software for viewing intricate details in the model. This makes MeshLab a better alternative for measuring accurately in the scan, as it is easier to spot the exact position of the control points. The distances were therefore measured by the Measuring tool in MeshLab. Measuring in MeshLab are shown in Figure 3.8 and 3.9.
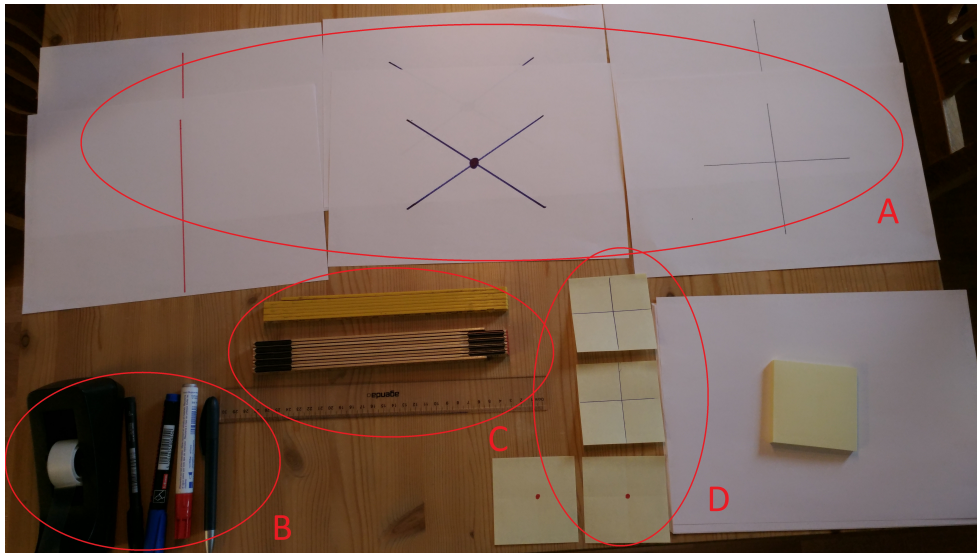


Figure 3.7: Equipment used for the test before starting.
A: Control points made from A4-sheets.
B: Markers.
C: Measuring tools.
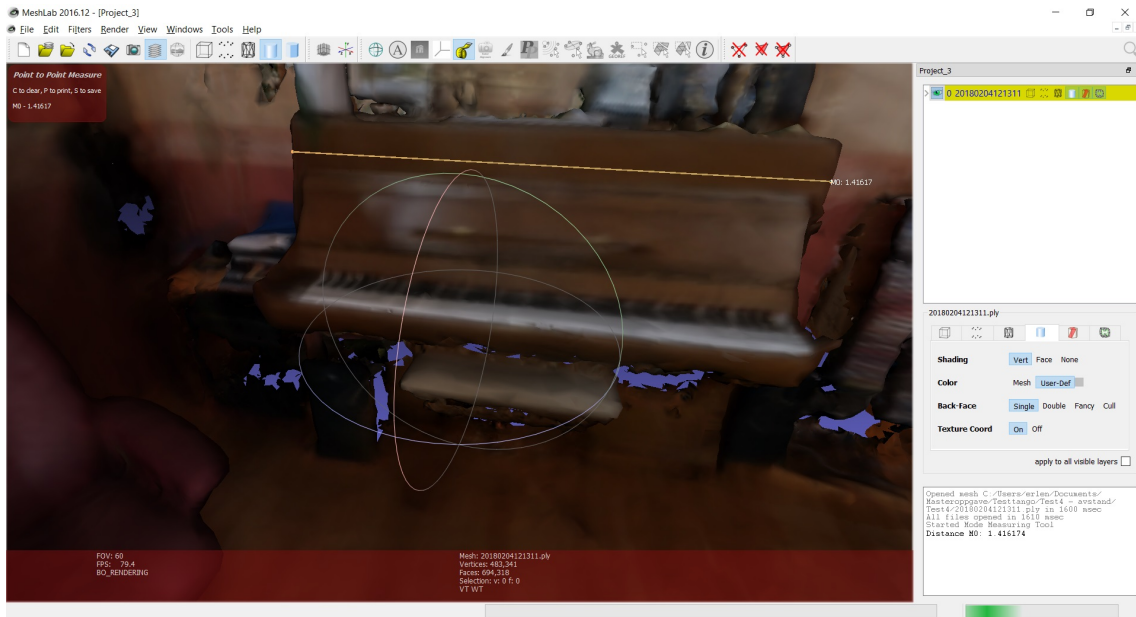D: Control points made from post-it notes.

Figure 3.8: Measuring the distance between two control points on an object in MeshLab.
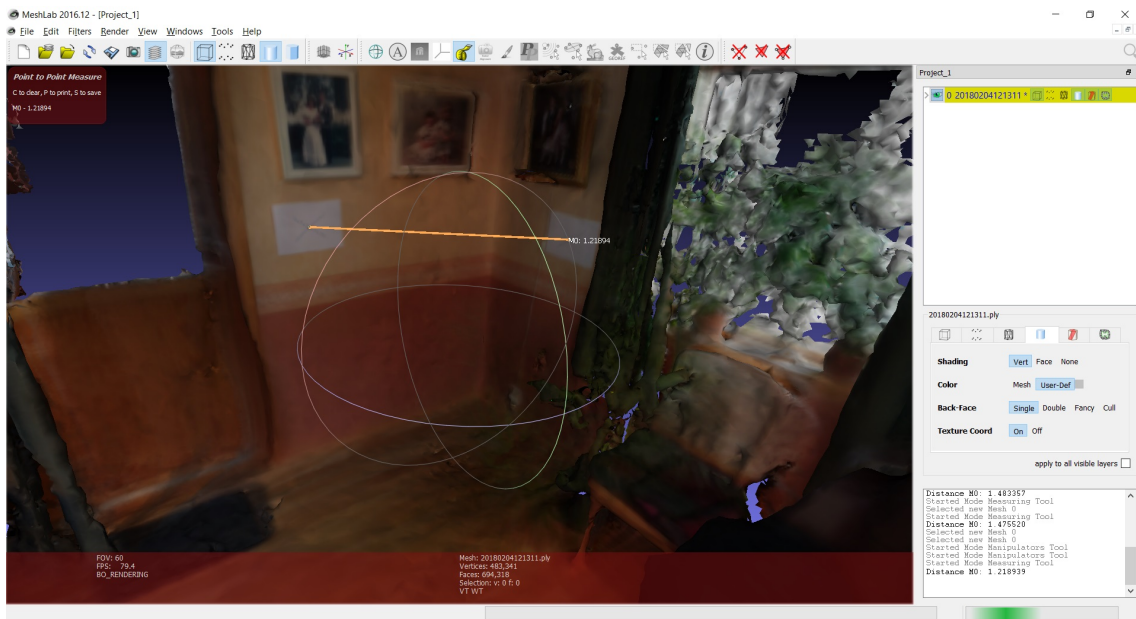


Figure 3.9: Measuring the distance between two control points made from sheets in Mesh-Lab.

The results of the measurements done by yardstick and by the measurement tool in MeshLab can be seen in Table 3.2. The distances in the real room were rounded up to the closest centimeter as accurate measurement was hard to obtain using the yardstick. The scanned measurements are shown with one decimal since it is easier to hit the accurate position in the software:

Table 3.2: Test 3, the results from measuring.

| | Distance in physical building | Distance in 3D model | Relative error |
|---|---|---|---|
| Control point A - Red Line | 148 cm | 147.5 cm | 0.5 cm |
| Control point B - Blue cross | 122 cm | 121.9 cm | 0.1 cm |
| Control point C - Black cross | 138 cm | 138 cm | 0.0 cm |
| Control point D - Red dot, Post-it notes | 100 cm | 100.0 cm | 0.0 cm |
| Control point E - Cross, Post-it notes | 193 cm | 193.3 cm | 0.3 cm |
| Control point *Piano* | 145 cm | 141.6 cm | 3.4 cm |
| Control point *Door* | 203 cm | 206.2 cm | 3.2 cm |
| Control point *Fire place* | 56 cm | 56.2 cm | 0.2 cm |

Lessons learned from the test:

- Control point solutions *A, B* and *C* are easily measured in MeshLab.

- The solution of control point *A* is inaccurate as it is possible to measure from several different locations on the line.

- It is hard to measure point *C* in MeshLab as the control points are on opposing surfaces.

- Post-it notes as control points is alright, but bigger sheets are better.

- It is possible to use objects as control points. Using objects often provides lower accuracy as it is hard to hit the exact measurement position in the 3D model.

- All over, the accuracy of the 3D model seems high, with a maximum of 0.5 cm inaccuracy on sheet control points and 3.4 cm on object control points.

**Test 4: Merging scans**

During the testing so far, the experience is that the scan app crashes occasionally, both during scanning and post processing. Scanning has to start all over again, which prolongs scanning time as well as making scanning of vast areas hard. As the experiment will be conducted on a huge building, a possible approach would be to divide the building into smaller parts, scan these separately and merge the scans during post processing. Thus, the goal of this test would be to find a feasible way of merging scans accurately.

Initially, several parts of the same building were scanned. Some parts were scanned to overlap each other in order to establish common merging points. The scans were sampled in Cloud Compare before transferred to ReCap where the individual scans could be transformed to a common origo by using the *Update origin*-tool. Figure 3.10 shows ReCap while updating the origin. If the origin as well as the coordinate axes of all scans are the

same, the scans should be perfectly aligned. If this is achieved, it would make it easy to merge several scans to one.
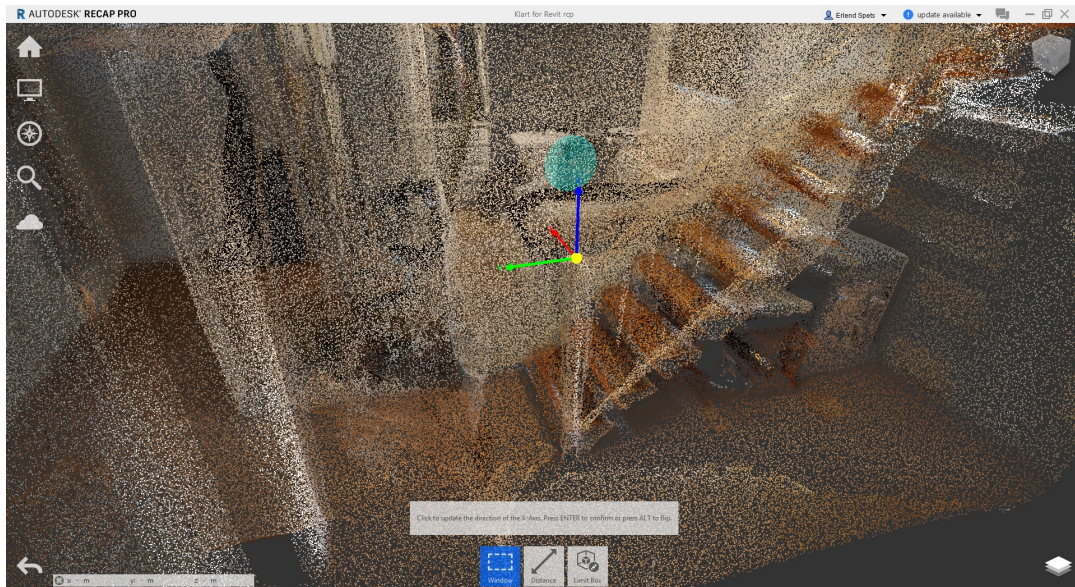


Figure 3.10: Point cloud alignment in ReCap. Including the origo and coordinate axes.

This method turned out to be quite inaccurate and face some problems when merging several scans, as all scans needed to use the same point as an origin. After discovering this, the search for another way to merge the scans in a more accurate way started. Cloud Compare also allows users to align point clouds by using the *Align two clouds-tool.* This is done by manually selecting similar points on two clouds, followed by an alignment algorithm. The algorithm automatically rotates and moves the clouds together based on the selected points. Cloud Compare also have an option for further automated alignment which can be initiated by clicking the *Finely registers already aligned entities* button. This algorithm is supposed to give even higher accuracy to the aligned point clouds. After testing this tool several times on the scans, the conclusion was that it gave a less accurate alignment than only using the manual tool. After the point clouds are aligned, they are easily merged by the *Merge multiple clouds-tool.* By aligning the clouds using the manual method only, it is possible to merge several point clouds by various points, making it unnecessary for all point clouds to include the same area. The merging is also quite accurate and gives the user the chance to merge all clouds before further processing in ReCap. The layout of Cloud Compare while merging is shown in Figure 3.11.
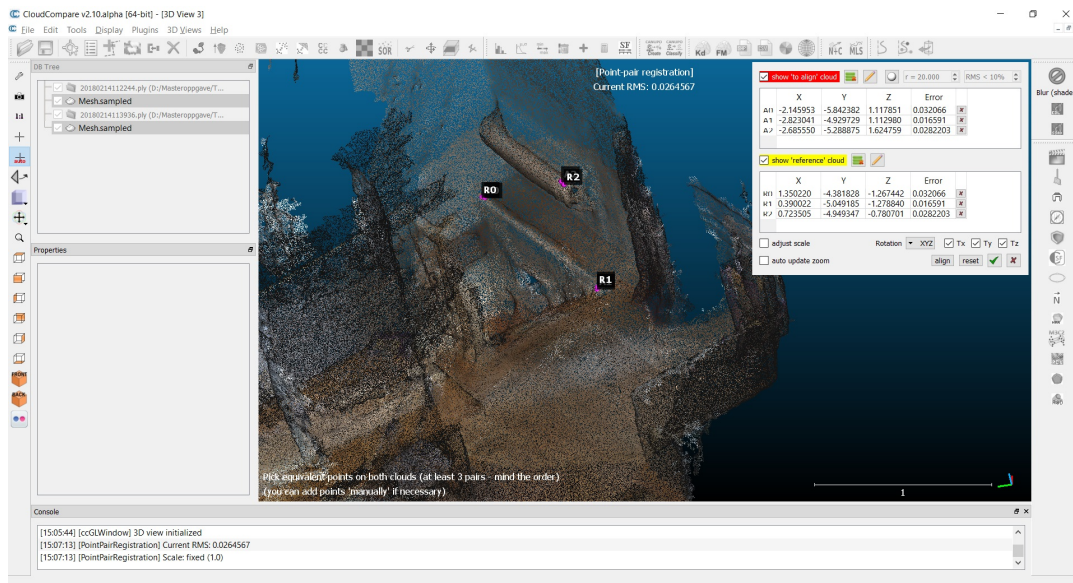
Figure 3.11: Point cloud alignment in Cloud Compare. Showing the two point clouds and the three alignment points after aligning.

Lessons learned from this test:

- Merging point clouds in Cloud Compare is easier and more accurate than merging in ReCap.

- Objects to be used as merging points have to be created. These points should be big enough to be visible in the scan, but small enough to be able to click on the exact same point in both scans.

- There needs to be big enough overlapping areas to make the alignment more accurate.

- The automatical alignment tool does not give higher accuracy when aligning point clouds.

- Avoid to many point clouds. Each merging will give less accuracy to the final 3D model.

**Test 5: Scanning the target building**

The final test includes a scanning of the target building. The purpose was to establish a scanning strategy with respect to scan partitioning. Hence, testing if it would be possible to scan the entire building in a single scan. Alternatively, testing merging points in a portioned scan and check if some unexpected issues would occur.

Before scanning, the building was checked and prepared for scanning. Merging points were placed in three central areas of the building as shown in Figure 3.12. These points would get usefull if scanning the building in a single scan would fail.

Figure 3.12: Merging points: Coloured cones were placed at central parts of the building during the test.

Scanning the entire building in a single scan proved to be difficult. The first scan lost tracking after 3 minutes, resulting in an error. The second scan lasted 37 minutes before an error occurred. During the entire testing, several errors occurred when scanning large areas, hence a decision was made to divide the building into 6 scans instead of a single scan.

Processing these 6 scans on the phone turned out challenging. The phone processing used between 30 and 60 minutes to reach 20-30%. An error occurred in the processing before it reached 30%. Since the phone was able to process smaller test scans, but not the large scans, it is likely that the error was caused by too large and complex scans. After about 10 hours of work, the scanning of the building was completed, resulting in 5 unprocessed scans and one processed scan. An unprocessed point cloud is shown in Figure 3.13
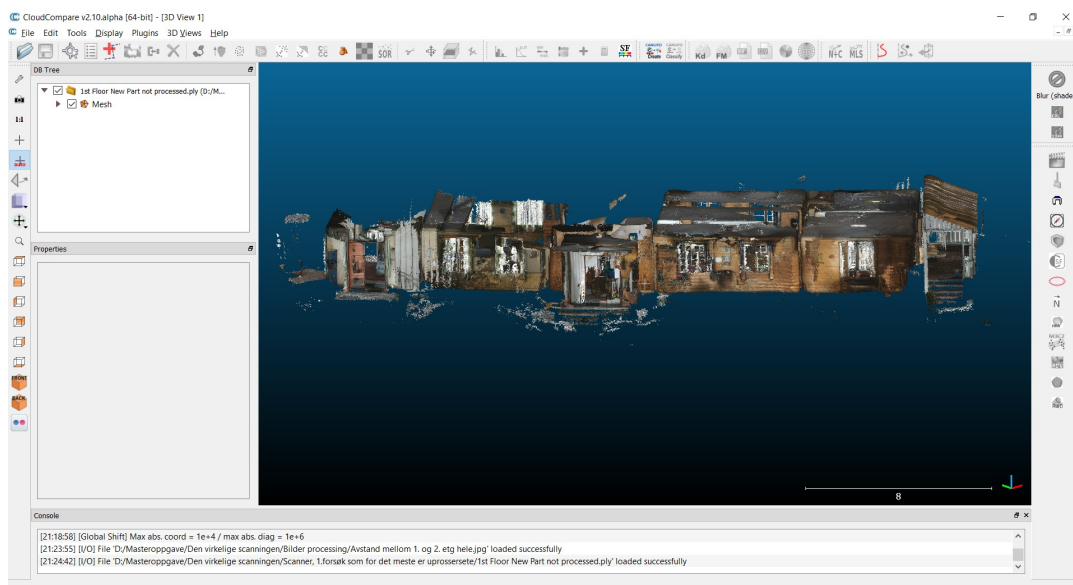


Figure 3.13: An unprocessed in Cloud Compare of the first scan during the test.

The 6 scans were transferred to a computer. During this process, it was discovered that the file size of the unprocessed scans was way bigger than the processed ones. A 35 minutes unprocessed scan had a size of 2.4 GB, while the size of a processed scan of 15 minutes was 150 MB. The unprocessed scans were hard to work on as the quality was bad and it was hard to spot the accurate position of the merging points. Still, it was possible to merge the unprocessed 3D models and obtain a point cloud suitable for further BIM creation. The test proves that this is a viable way of creating a BIM from smartphone 3D scanning, but it would be more feasible to separate the building into smaller scans. This would make the constructor app able to complete its post processing.

Lessons learned from the test:

- Smooth surfaces without much variation increase that chance that the app loose tracking. These surfaces would have to be scanned slowly.

- Too dark areas can make the scan loose its position in the coordinate system. If this happens, the app starts to overwrite areas already scanned in a slightly different position. To dark areas must be avoided.

- If there is an interruption on the cell phone, like an incoming call, the software might malfunction, and the scan will be lost. This may be avoided by switching the phone to flight mode.

- After a failed scan, there are some times a need to restart the phone before continuing the scanning. Having many failed scans will drastically increase the time spent on scanning.

- The Constructor app is unable to process big scans. Scanning 1-4 rooms each time would be a more feasible solution.

- The merging points made for this test were too small, it would be better with bigger cones to make alignment easier.

- It is hard to hit the correct points for merging in Cloud Compare if the entire room is scanned, because the walls encircle making it hard to hit the right points. A better solution would be to scan half the room with the merging points, to make alignment easier.

## 3.2 Experimental Setup

### 3.2.1 Location



Figure 3.14: The Farmhouse viewed from south-east.

The chosen building, which is displayed in Figure 3.14, is a farmhouse located in Melhus, Norway. The age of the farmhouse is unknown, but it is assumed to be over 250 years old. A photo of an inscription on a roof is shown in Figure 3.15. During its lifetime, the house has been both extended and rebuilt. There are no drawings describing either the structural system of the building or the design. As the farmhouse is made from wood and has endured a long lifetime, some beams and floors are sagging, and some walls are bent. It is also a big house, about 34 m long and 7 m wide, having two main floors as well as a cold attic and a cold basement, with all floors containing different kinds of rooms and interior. Some parts are modern residence flats, while other parts have not been used for decades. In total, the house is a realistic object for a renovation, demolition or rebuilding project. This building provide multiple challenges for exploring smartphone Scan to BIM.

Figure 3.15: *1762*, is written as an inscription on top of one of the farm buildings. This gives clear indications that the building is over 250 years old.

### 3.2.2   Equipment

The equipment used for this experiment will be divided in the following sub categories:

**Scanning equipment:**

**ASUS Zenfone AR:** Is the main scanning equipment used. The cell phone has an Android operative system. The phone's specifications are shown in Figure 3.16 [54].

- 128 GB Storage capacity
- Li-Ion 3300 mAh battery
- TriCam 23 MP Camera
- 6 GB RAM
- Adreno 530 GPU
- Quad-Core 2x2,35 GHz  2x1,6 and GHz CPU
- Snapdragon 821 Chipset
- Weight: 170g
- GPS and GLONAS trackingsystem



Figure 3.16: ASUS Zenfone AR with specifications [54].

The phone was unveiled in January 2017 and released in June 2017. It was the 2nd phone ever launched containing the Google Tango platform. Given the phone's current hardware it is the best cell phone with Google Tango on the marked, making it an obvious choice for smartphone 3D scanning.

**Phantom 4 Pro Drone:** Was used for scanning the exterior of the building from the air. The drone has an integrated mounted camera which allows the user to view and photograph from the drone's position. The Drone is easily controlled by a hand held controller. The Drone's specifications are shown in Figure 3.17 [55].

- Size: 350mm x 350mm
- Max controller distance: 7km
- Take off-wight: 1388g
- Max Horizontal velocity: 20m/s
- Max Vertical velocity: 6m/s
- Max Angular velocity: 250°/s
- 20 Megapixel camera
- Maximum flytime: 30 min
- Max storage: 128 GB



Figure 3.17: Phantom 4 Pro with specifications [54].

**Measuring equipment:**

**Leica DISTO X310:** Was used for measuring the physical distances between the control points before scanning. The specifications of the small, hand held laser measuring device is shown in Figure 3.18 [56].

- Certified typical measuring accuracy: +/- 1mm
- Certified typical measuring range: Up to 120m
- Dimensions: 122mm x 55mm x 31mm
- Weight: 155g
- 360° Tilt Sensor
- Memory to Store 30 Displays



Figure 3.18: Leica DISTO X310 with specifications [56].

**Sheets with markings:** Were used as control points. Regular A4 sheets and post-it notes were marked with an aim and put on the walls, floors and ceiling inside the house as control points. Figure 3.19 shows two marked sheets used in the experiment.



Figure 3.19: Marked sheets which was used in the experiment.

**Other equipment:**

**Lenovo Ideapad 700-15ISK Signature Edition:** Was the computer used for handling and saving the scanned data as well as post processing and BIM creation. The laptop is running on a Windows 10 operating system. The laptop's specifications are shown in Figure 3.20 [57].

- Dimensions: 384mm x 265mm x 22,7mm
- Weight: 2,3kg
- Processor: 6th Generation Intel Core i5-6300 HQ CPU, 2.3GHz
- Chipset: Intel HM170
- RAM: 16GB
- Graphics: Intel Graphics 530 in processor and NVIDA GeForce GTW950, 4GB memory
- Storage: 1TB 5400RPM and 128GB PCIe SSD
- Display: 15.6", 1920x1080 resolution



Figure 3.20: Lenovo Ideapad 700 15" [57].

**Cones:** Were used for aligning and merging scans in the post processing. Made from coloured A4 sheets as shown in Figure 3.21. These cones are easily spotted in the scan and are clearly defined points, well fitted for merging.



Figure 3.21: The Cones used in the experiment

### 3.2.3 Pre-scanning setup:

The following preparations were done before the start of the experiment:

1. The entire building was checked to make sure it was ready for scanning. Lights were turned on, doors were opened, and the scanning route was planned.

2. The Control points were made by marking sheets of paper with a definite "X" for measuring. Such a control point is shown in Figure 3.22. These were fastened by tape to surfaces of the building. Five control point pairs were put on opposing walls, five were put on floors/ceilings, five were put parallelly on the same walls while five were put on perpendicular walls. To make the measurement accurate, the control points were checked to see that they were placed directly opposing each other as shown in Figure 3.23.
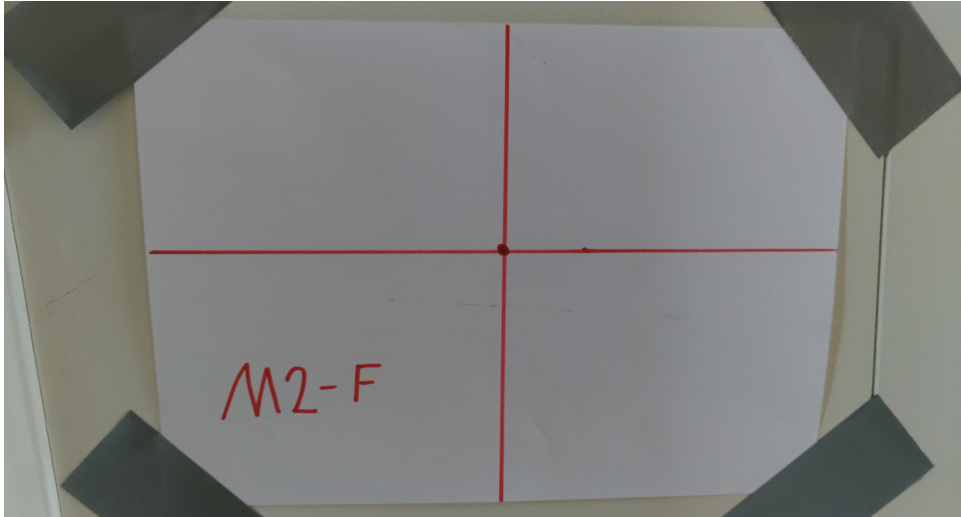
Figure 3.22: Control point M2-F



Figure 3.23: Checking that the control point was placed directly opposing the other during experimental setup. The Leica measuring device was placed in control point A, and it was controlled that the laser beam hit the center of control point B

3. The distances between the control points were measured and recorded, as displayed in Figure 3.24 and 3.25. Each distance was measured twice. If there was a 1 mm difference between the two results, a third measure was conducted and the result with two similar measurements were recorded. If there was more than 1 mm difference between the measurements, two new measurements would be conducted. It turned out that it was hard to measure the distances on perpendicular walls. This was caused by the fact that the Leica Ditance Measurer had to be placed at an angle to the wall, causing a lot of error on the measurements. It was then decided that the perpendicular measurements would be dropped.
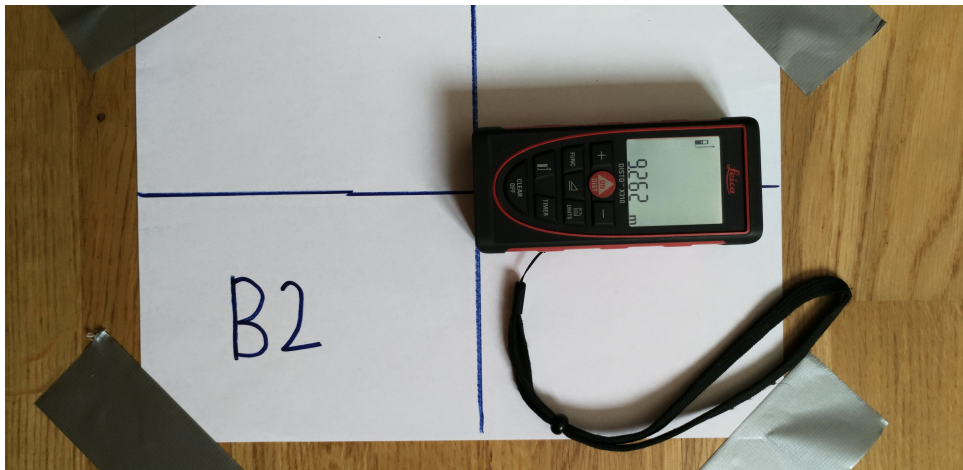


Figure 3.24: The Leica DISTO X310 was placed with the rear end at the vertical line with the laser pointer just above the horizontal line to make the measuring as accurate as possible.



Figure 3.25: The control points placed parallelly on the wall were measured by placing a box on the control point at the right spot.

4. The cones to be used as merging points were created. They were made larger than in the final test so that they could easily be spotted in the model. A total of 10 merging cones were made. Five colours were used with two cones in each colour.

5. The cones were put in position on two merging areas as shown in Figure 3.26



Figure 3.26: The setup of the first merge point spot.  The red circles show the five merging points.

6. The drone, which was to be used for exterior scanning, was tested and the battery was fully charged. The drone testing is shown in Figure 3.27



Figure 3.27: Checking the drone before starting the experiment.

### 3.2.4  Plan for scanning and processing:

An overall and a detailed plan for the experiment and the various parts were created on the basis of all tests that were conducted.

**Overall plan:**

1. Scan the interior of the building. Do this on a couple of rooms at a time, such that each scan will not last more than 15 minutes. After each scan, allow the internal processing on the phone to finish before starting on the next scan, to make sure that the phone will finish the processing.

2. Scan the exterior by using a drone with a camera. Start processing the material from the drone scan and create a 3D model of the house exterior.

3. Export all the scans to a Google drive so it can be accessed by a computer and check that all scans are working properly.

4. Measure the distance between the control points in MeshLab and note the results in a table.

5. Sample, clean, align and merge all the scans in Cloud Compare. Export the merged scan as an *E57* point cloud file.

6. Open the *E57* file in Autodesk ReCap. Create regions of the point cloud for floors, walls, ceilings etc. Export the model as a text*itrcp* file.

7. Import the *rcp* file as a point cloud in Revit and use this to make a BIM.

**Detailed plan for specific areas:**

- **Time of day when scanning:** Test 2 shows that the time of day when doing the scan does not have much influence on scan quality if the area is well lit. In this old farmhouse, some of the rooms have poor or non-existing lights, meaning that it would be preferable to scan during daylight. Hence the plan would then be to do all the scanning in daylight.

- **How to scan rooms:** To scan rooms well and efficiently, it would be feasible to scan every room as shown in Figure 3.28. Since most rooms contain obstacles, scanning exactly like this is seldom possible and some improvising would be necessary. It is possible to see what is being scanned while scanning, which makes it possible to check that everything is scanned from all angles before leaving a room. In dark or monotone areas with few reference points, strive to scan slowly and with some distance from the phone to the scanned area.
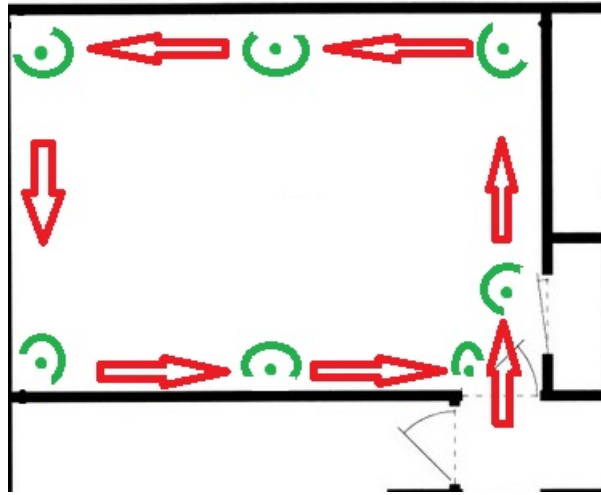
Figure 3.28: Optimal room scanning:  The red arrows show how to move in the room, the green dots show where to stop and the green arches show which directions should be scanned from that specific area.

- **Drone scan and drone scan processing:** The drone scan consists of flying the drone around the house while taking photos from various angles. The plan would be to do circles around the house at various altitudes. Photos will be taken at approximately every 2-4 meters. After completing this, the photos will be exported to *Pix4D*, where an automatic photogrammetry processing will take place to create a 3D model. The plan will be to choose the option of highest resolution and clear all noise from the model before exporting it as a *las* file.

- **Noise removal:** As every room scan contains a lot of unintended areas and points, there is a need to clean the scan before merging. These unwanted points are for example coming from scanning mirrors, as the phone thinks that the room continues inside a mirror because of the reflection. Cleaning will be done by using the *Cross section* tool in Cloud Compare. This will be done by moving the section box around the model so that just the intended scan is left inside the box. Then creating a new section of this area and delete the old one. The process is shown in Figure 3.29 and 3.30
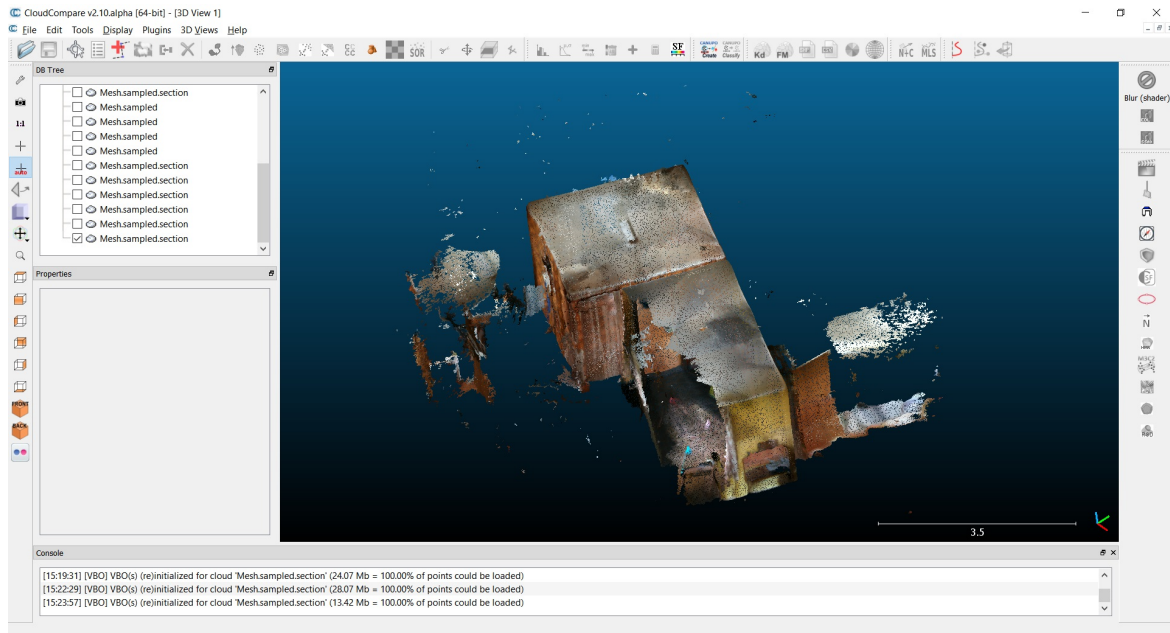
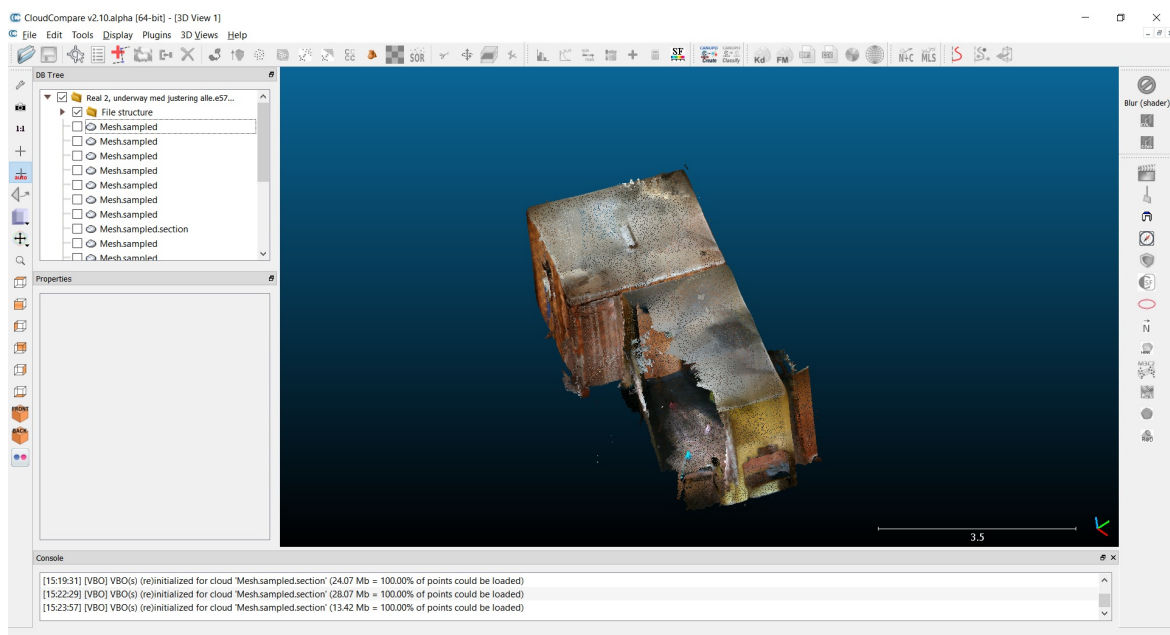Figure 3.29: Cloud Compare cleaning: Before cleaning the point cloud.



Figure 3.30: Cloud Compare cleaning: After cleaning the point cloud.

- **Merging plan:** The plan is to use six different merging areas while scanning the entire building. Each merging area will consist of five cones which were prepared for the experiment. The distance between the cones will be as large as possible to make the merging as accurate as possible. When scanning areas around the cones, make sure not to scan all the surrounding walls. This is done to make it easier to select the cones while processing. Scanning the areas connected to each merging point will happen accordingly:

1. Start by scanning all areas around the first merging area.

2. Next, do a *connecting scan* by scanning from the first merging areas to the second. An example of a *connecting scan* is shown in Figure 3.31. The focus of this scan will be to scan the cones in both areas well to make alignment and merging easier.

3. When finished with the first scanning area, move the cones from the first to the third merging area.

4. It is then time to scan all areas connected to the second area.

5. When that is done, repeat step 2-4 from area 2 to area 3 and continue doing this until the entire building is scanned.
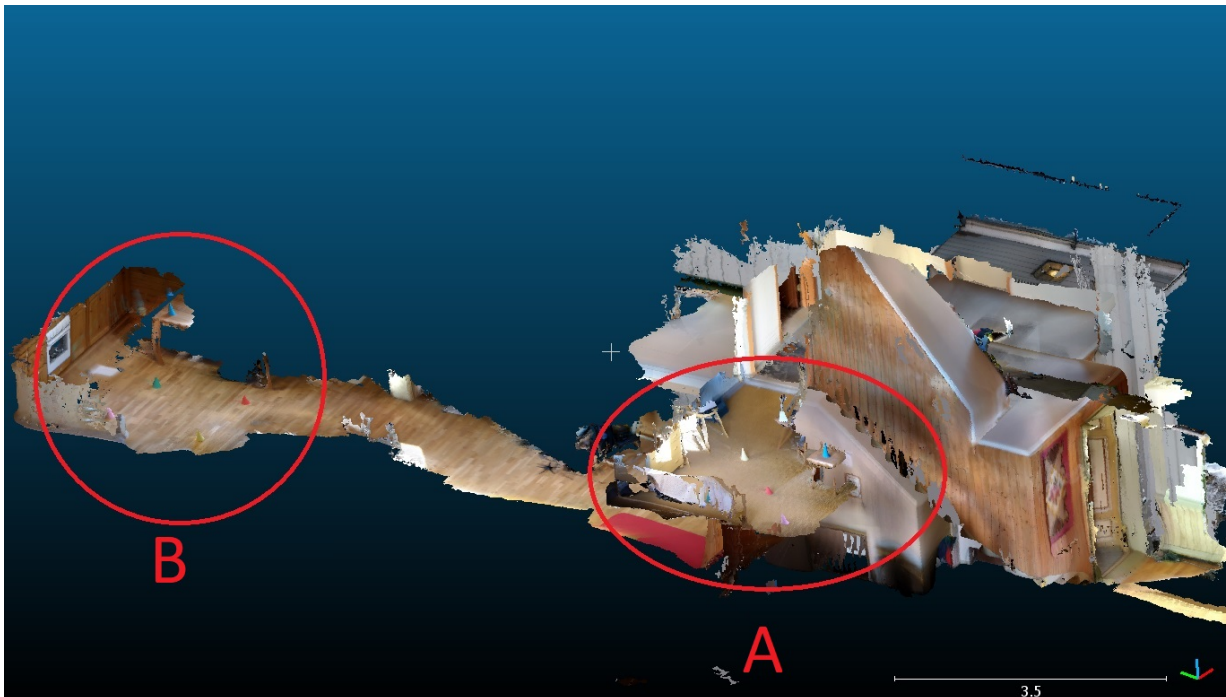


Figure 3.31: A *connecting scan* between two merging areas. This is done after all scans connected to area A is finished and it is time to scan areas connected to area B.

- **Merging processing plan:** When cleaning of all the scans in Cloud Compare is finished, it is time to merge the scans into one. This will be done by using the alignment and merging tool in Cloud Compare in the following way:

  1. Two connecting point clouds will be aligned by choosing the top points of the five merging cones which are similar in the scans.

  2. The next connection point cloud will be aligned to the two first ones using one of them as a reference while aligning.

  3. Continue aligning the rest of the point clouds by using the already aligned clouds as reference.

  4. Then watch the entire model to check that everything is aligned as it should, before using the *Merge multiple point clouds*-tool to merge all point clouds into one.

- **Creating regions:** This will be done in Autodesk ReCap after the scans are merged into one. It will be done by using the *Plane*-tool. This will provide the option of targeting all points with a 0-100 mm distance from a chosen plane. The plane is chosen by clicking on several points on the model. When a plane is chosen, it is possible to save it as a region with a name describing it, lock it for further editing, hide it and then choose a new region. This will be done with all exterior walls, floors, ceilings and roof. The rest of the interior walls, obstacles and exterior will be stored in a region called *rest*.

- **BIM creation:** As there are no automatic way of generating a BIM from the model, this will have to be done all manually. Autodesk Revit is the designated BIM software. The point cloud should be imported and used as a template for further BIM creation. The first step will be to create levels, views and grids, fit for the building. When this is done, drawing can start. At first, the plan will be to make straight lines fitting on the edges of objects like walls, windows etc. These lines will be used to create objects fit for the model, including wall thickness, window and door size, floor thickness and more. After the families are created, it is possible to start placing the objects and connecting them in an appropriate spot. The ground and terrain around the building will be measured approximately in ReCap, before starting to place curvature points. These points will be checked against the point cloud to gain an accurate elevation around the building. Some improvisation and training will have to be done during Revit BIM creation.

## 3.3   Execution

### 3.3.1   Scanning

The interior of the house was scanned in a stepwise fashion. A few rooms were scanned in each step. Six scans had to be repeated due to malfunction of the scanning app or the tracking function. These were later deleted. The successful scans of the house interior are shown in Table 3.3 and 3.4:

Table 3.3: Scan 1-10.

| Scan number: | Area | Scanning time | Phone processing time | Comment |
|---|---|---|---|---|
| 1 | One room and hallway, 2nd floor of the southern part of the building. | 7 min | 21 min. | Tried to scan 3 rooms at first, but processing crashed. |
| 2 | Two rooms, 2nd floor of the southern part. | 8 min | 29 min | *None* |
| 3 | Two rooms and rest of the hallway, 2nd floor of the southern part. | 10 min | 22 min | Processing crashed during the first time scanning this part. |
| 4 | Connecting scan, from the 2nd to 1st floor. The entrance hall of the southern part were also scanned. | 11 min | 19 min | *None* |
| 5 | Two rooms, living room and kithchen in the end of the southern part. | 13 min | 20 min | Lost tracking at the end of the scan, but managed to acquire everything needed. |
| 6 | Three Rooms of the 1st floor, southern part and the rest of the living room. | 15 min | 25 min | Tried to scan 3 rooms at first, but processing crashed. |
| 7 | Connecting scan to new merge area in the middle room of the house. | 7 min | 10 min | *None* |
| 8 | The two rooms in the middle of the house, 1st floor. | 13 min | 42 min | The app crashed twice while doing this scan, 1st time because of lost tracking, 2nd time by error while processing. |
| 9 | Connecting scan to new merge area on the 2nd floor of the northern part and half a room. | 10 min | 30 min | The phone crashed the first time doing this scan after one hour of processing. |
| 10 | One room, the hallway and the balcony on the 2nd floor of the northern part. | 14 min | 16 min | *None* |

Table 3.4: Scan 11-17.

| Scan number: | Area | Scanning time | Phone processing time | Comment |
|---|---|---|---|---|
| 11 | Connecting scan to new merge in the area 2nd floor. Scanning one room plus another hallway on the 2nd floor of the northern part. | 10 min | 21 min | *None* |
| 12 | One room and the last part of the hallway in the 2nd floor of the northern part. | 9 min | 8 min | *None* |
| 13 | Two rooms on the north end of the 2nd floor. | 13 min | 23 min | *None* |
| 14 | Connecting scan from the 2nd floor to a new merging area on the 1st floor of the northern part. Scanned the entrance hall and the exterior around the exit under the balcony. | 14 min | 22 min | *None* |
| 15 | Two rooms plus the exterior and exit on the northern part of the 1st floor. | 7 min | 10 min | *None* |
| 16 | Scanned the kitchen on the 1st floor of the northern part. | 5 min | 8 min | *None* |
| 17 | The rest of the kitchen plus the living room of the 1st floor in the northern part. | 15 min | 35 min | *None* |

When viewing the scans on the phone, it was noticeable that it could be hard to see all the control points clearly. Hence, a decision was made to preform extra scanning of some of the control point areas. This was done in scans only targeting the control points.

The exterior was scanned quite quickly. The drone used 15 minutes to complete the entire scan. It flew three times around the building, one time over the top and did some extra photos around the entrances to make merging easier. The result was 137 photos of the building from various angles.

### 3.3.2   Processing

The scanned files were uploaded from the phone to a Google drive, making them available for post processing on a computer. After the files were organized on the computer, the post processing can be divided into three primary areas:

**MeshLab measurements:** The scans dedicated for distance measurements provided control points which were easy to see and quite easy to target accurately in MeshLab. The inaccuracy when measuring the distance in MeshLab lies with hitting the exact right point with the pointer. Figure 3.32, 3.33 and 3.34 show measuring in MeshLab:
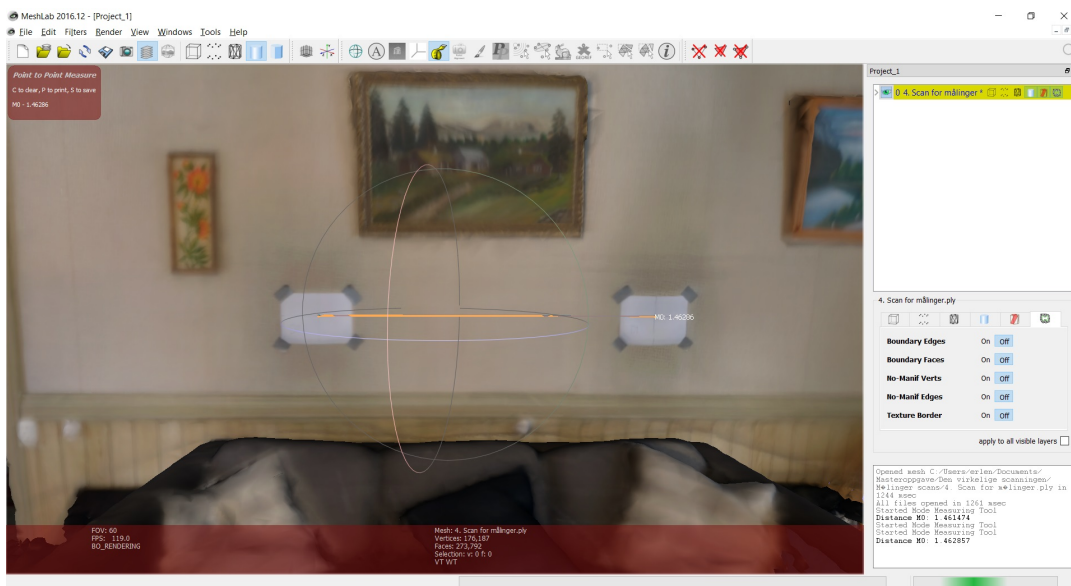


Figure 3.32: Distance measurement between two parallel control points on a wall.
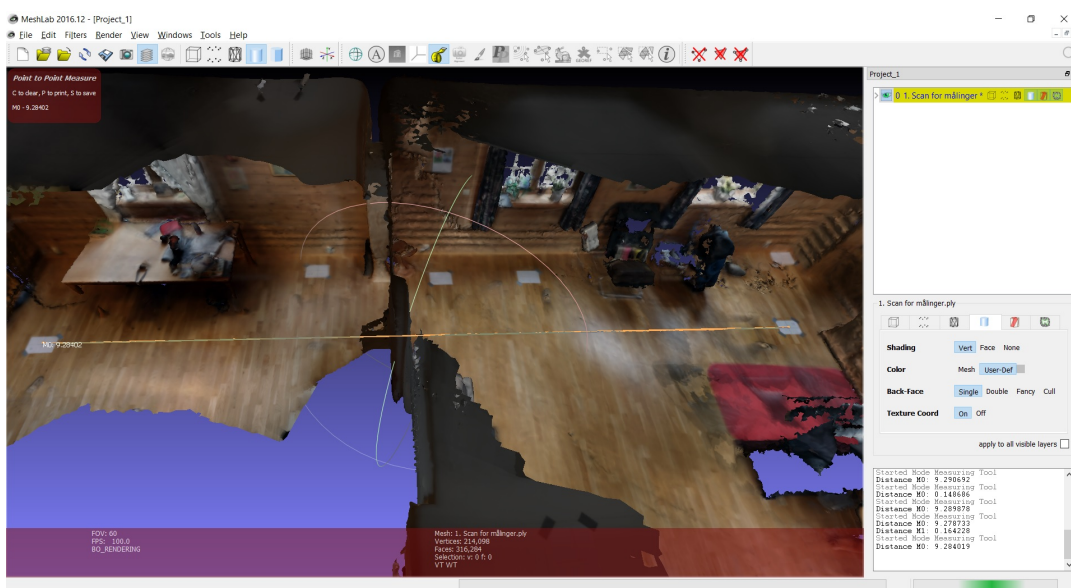


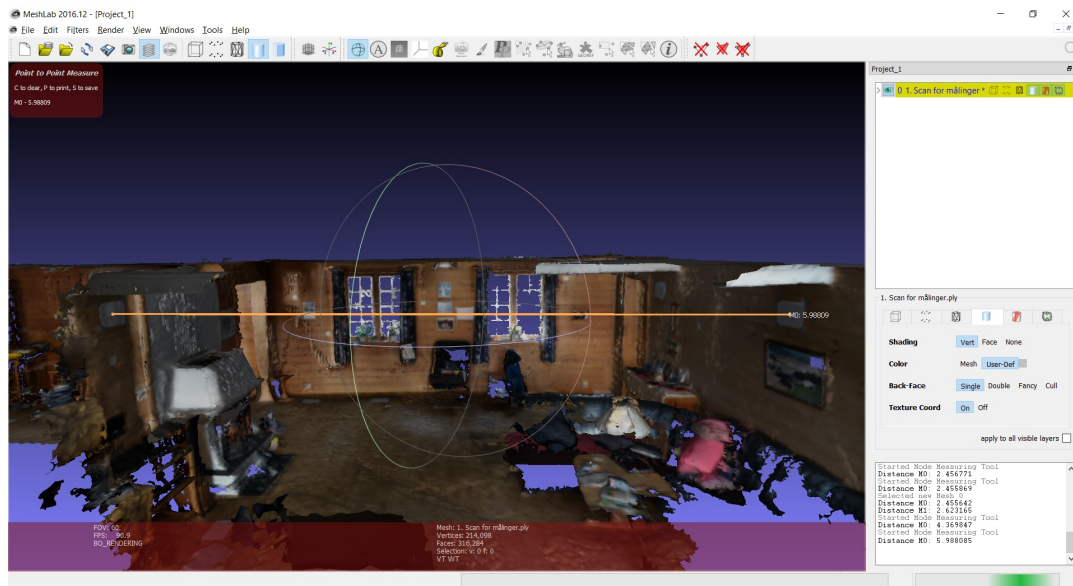Figure 3.33: Distance measured between two parallel control points on the floor.

Figure 3.34: Measure the distance between two control point on opposing walls.

**Cloud Compare sampling, aligning, cleaning and merging:**

The exterior scanning by the drone resulted in a *las* file containing a point cloud of 9 million points, while the interior scanning with the phone and constructor app provided a *ply* file containing a mesh. The *ply* files had to be sampled into point clouds in Cloud Compare before merging the exterior and interior scans. The processing in Cloud Compare was done in four steps:

1. **Sampling:** To create a uniform density in the final point cloud, sampling was done using cloud density and not total number of points as a reference. A density of 10 000 points was chosen, giving most scans 1-3 million points.

2. **Aligning:** The first step was to align interior point clouds. During this work, some clouds did not fit the rest as they should. This made the house look more bent than it is in reality. When all interior clouds had been aligned, the exterior was aligned by picking 2-4 points on each entrance area. As expected, the scans that looked bent made some parts of the interior cross the boundary of the exterior as shown in Figure 3.35. The exterior was also a bit rotated compared to the interior. Manual rotation of two interior scans and the exterior fixed this problem. The cause of this problem was probably badly scanned merging points. Figure 3.36 shows how it looks during interior alignment, while Figure 3.37 and 3.38 show how it looked like when aligning the interior with the exterior.

Figure 3.35: Checking error in alignmen.
Area A: The interior is crossing the boundary of the exterior.
Area B: There is a bigger gap between exterior and interior scans than for the rest of the model.
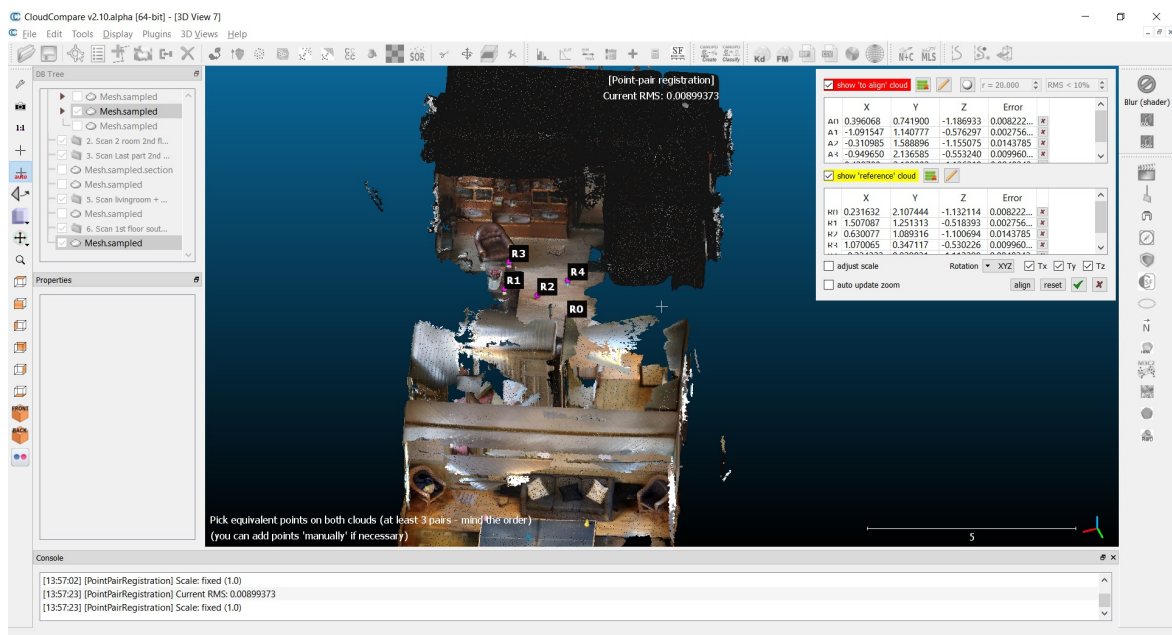


Figure 3.36: Aligning two point clouds on the 1st floor of the Southern part of the building with high precision.
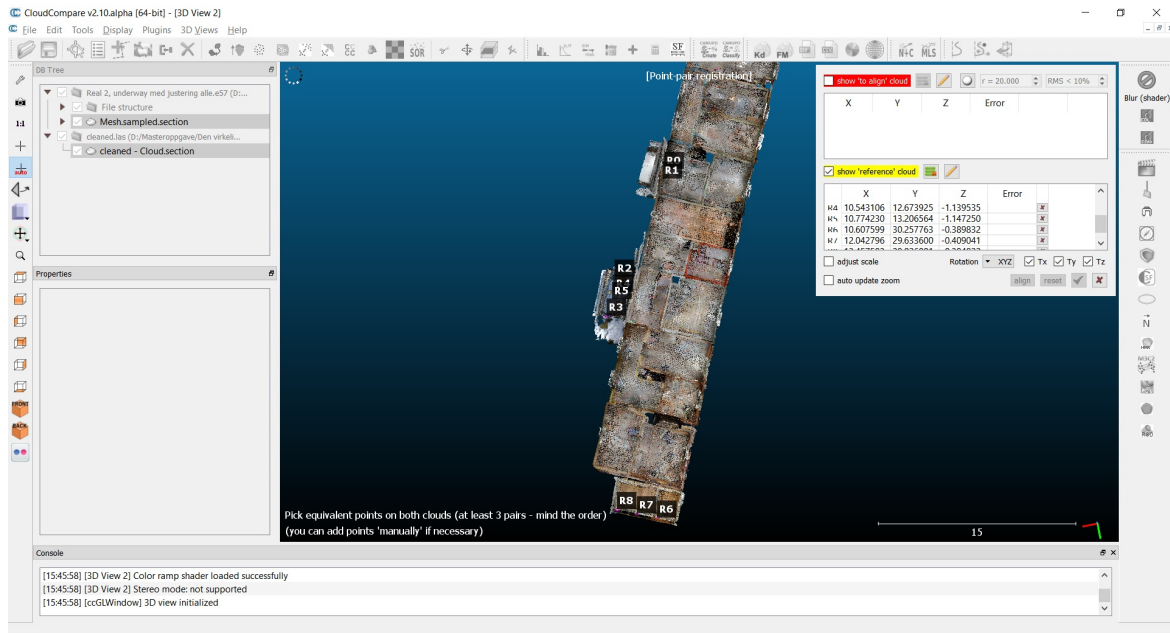
Figure 3.37: Aligning the exterior and the interior: Choosing alignment points on the interior constructor scans.
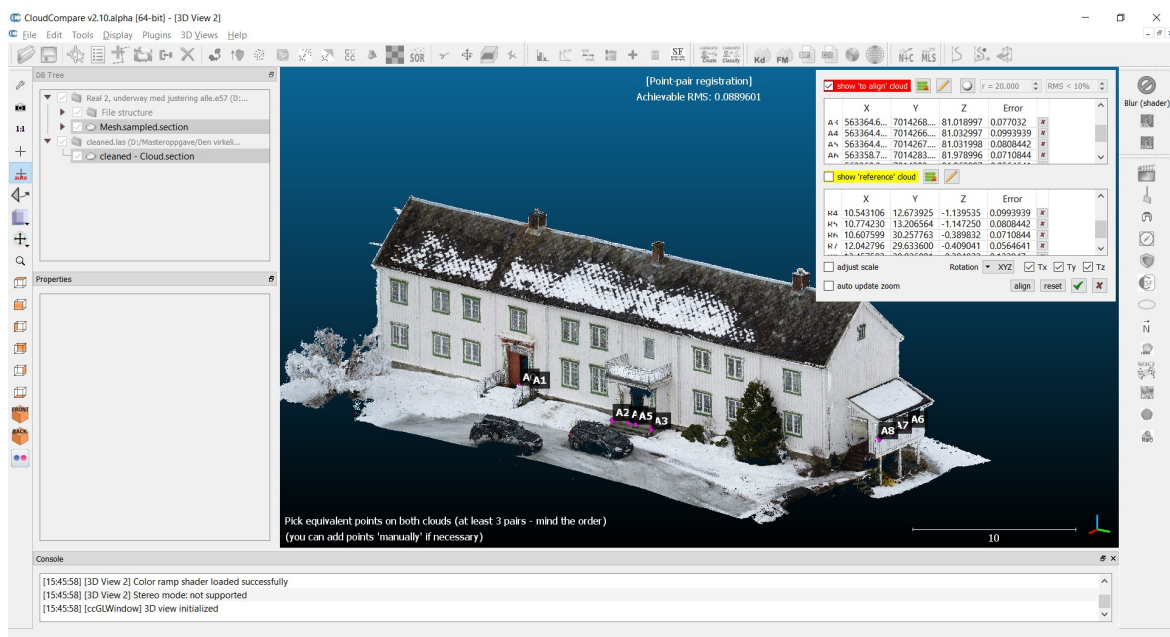


Figure 3.38: Aligning the exterior and the interior: Choosing alignment points on the exterior drone scan.

3. **Cleaning:** By cleaning the point clouds after they were aligned, it was possible to delete points in overlapping areas at the same time as deleting the other noise. Hence, a decision was made to perform cleaning after aligning. As Figure 3.39 and 3.40 shows, it was possible to remove a lot of points by cleaning this way, causing the file both to work smoother and look better.
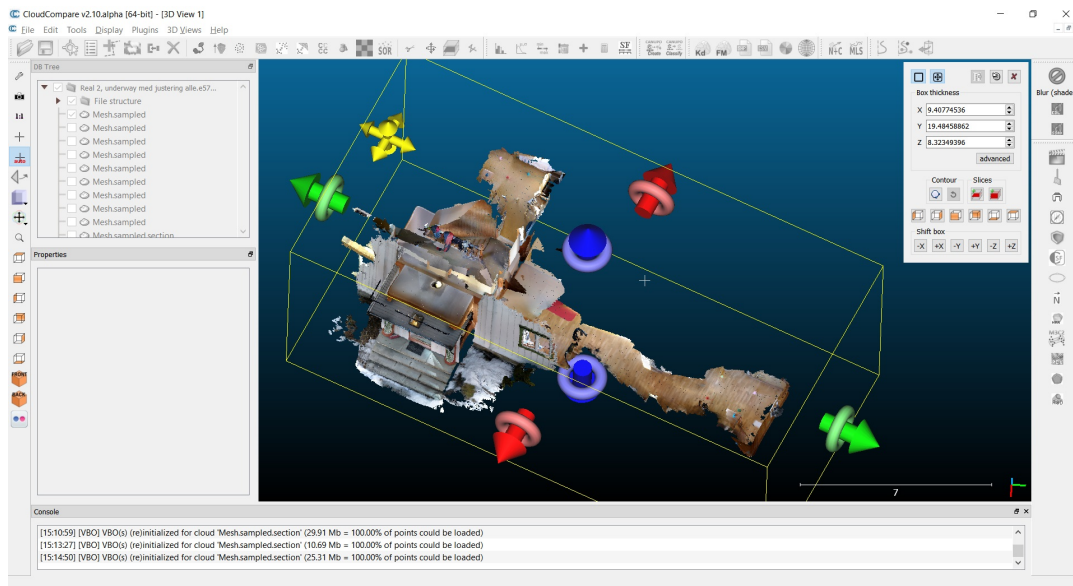
Figure 3.39: The scan before cleaning: Doing the cleaning after alignment made it possible to remove overlapping areas.
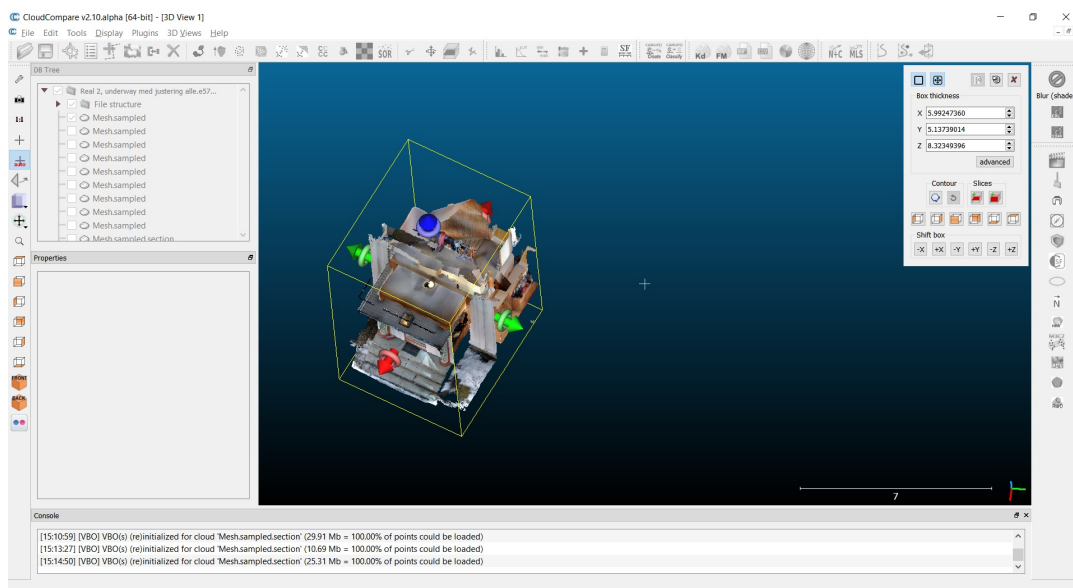


Figure 3.40: The scan after cleaning: Doing the cleaning after alignment made it possible to remove overlapping areas.

4. **Merging:** After aligning and cleaning, the result was checked to see that everything looked smooth. Merging the scans were done by using the, *Merge Multiple Clouds*-tool. All point clouds before merging are displayed together in Figure 3.41. The merged point cloud was saved as a complete 3D model in a *E57* file and exported to ReCap. The final *E57* file contained a total of 39 892 274 points.
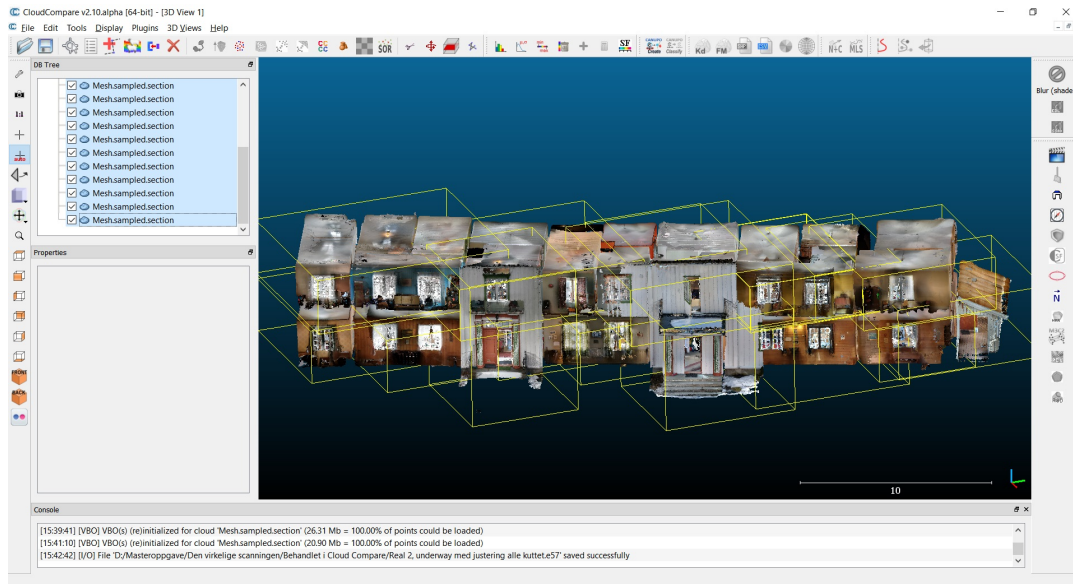
Figure 3.41: Could Compare: All interior point clouds aligned and ready to be merged.

**Autodesk ReCap final cleaning and region creation:**

The limit box in ReCap enables checking of the 3D model for missing parts. After checking the model, region creation was initiated. Doing it by selection points on a plane turned out troublesome as the surfaces were more bent than expected, making the tool only select parts of the intended surface. This is shown in Figure 3.42. Only a few regions could be created this way. The rest of the regions were made by making the limit box as small as possible around one area at a time and creating a region out of the selected area. This worked out quite fine, and it was used to create regions of all intended areas, making a total of 13 regions. The file was then exported as a *rcp* file, ready for Revit BIM creation.
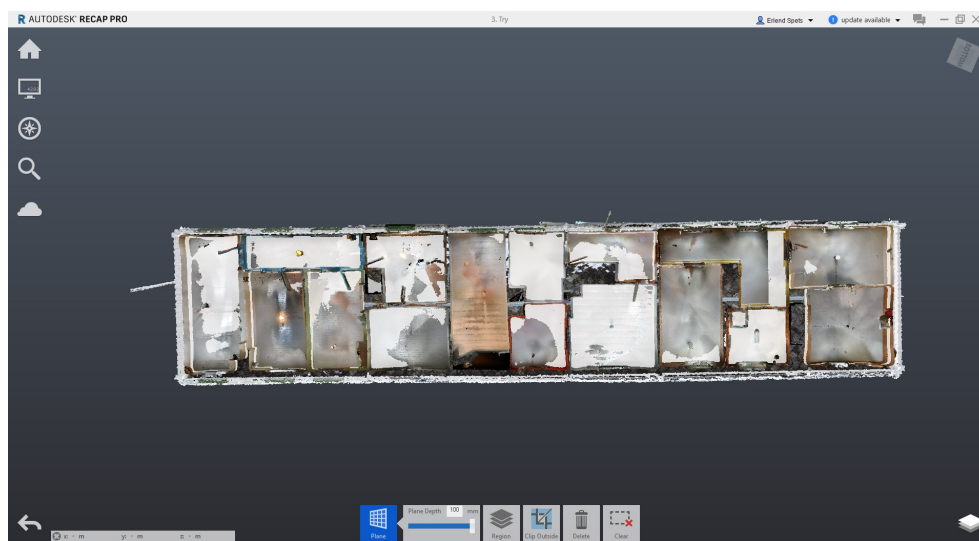


Figure 3.42: Region creation: The white areas show targeted points. Even when choosing 100 mm distance from the plane, only a part of the floor is selected.

### 3.3.3   Creating a BIM

The work in Revit started by creating a project with an architectural template and importing the *rcp* file from ReCap. After importing the point cloud, Revit started lagging and it was hard to use 3D view as the software did not run smoothly. Revit has the option of hiding point clouds. As the imported point cloud was divided into several regions, specific parts of the point cloud could be hid while designing. By hiding parts of the point cloud during BIM creation, it was possible to operate in Revit more smoothly. The BIM creation was done in the following order:

1. **Views, levels and grid:** First, sections and levels were created using a visual estimate of the best fit. Levels were created for all further working planes. As the entire first floor was not on the same level, some specific floors were created for these areas. Sections were made for each side of the building. Figure 3.43 shows the levels created by best fit. A grid system of the walls was also made during this stage.
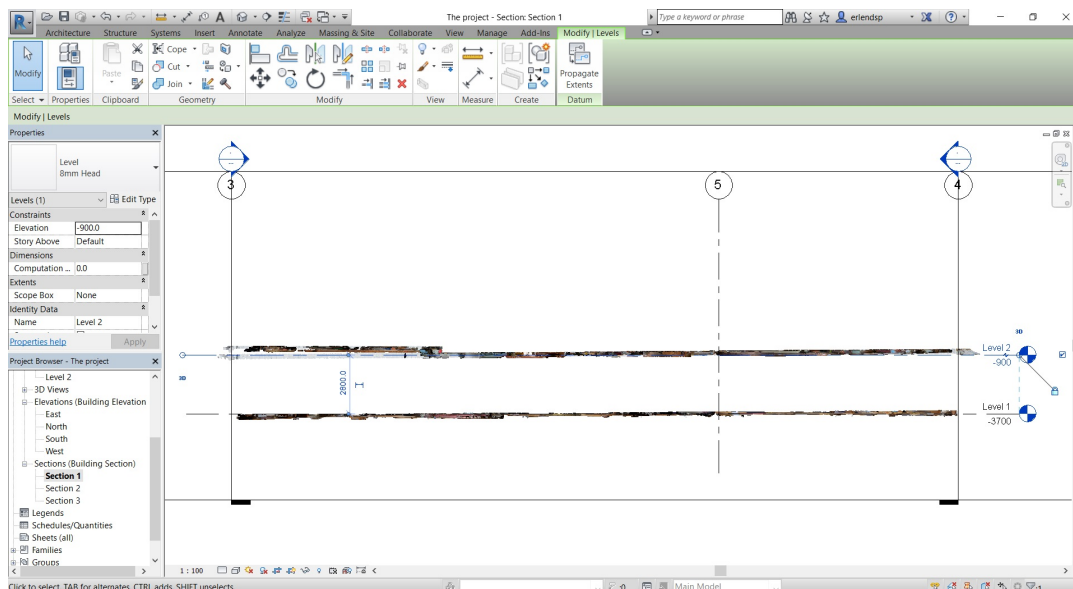


Figure 3.43: Revit during BIM creation: Adding level 1 (Ground floor) and 2 (First floor) by looking at the point cloud from a side section.

2. **Creating the exterior walls of the house:** The walls were made by first drawing a approximation of straight lines over the exterior walls. As the exterior model was not aligned perfectly with the interior, the thickness of the walls was varying across the model. In order to make the exterior walls of the BIM with continuous thicknesses, the average value of the walls thicknesses were calculated. Some of these measurements are shown in Figure 3.44. The calculated thickness was used when creating a Revit exterior wall family. To make the outer part of the BIM wall look realistic, a photo of the real exterior walls was used as the appearance of the wall material.
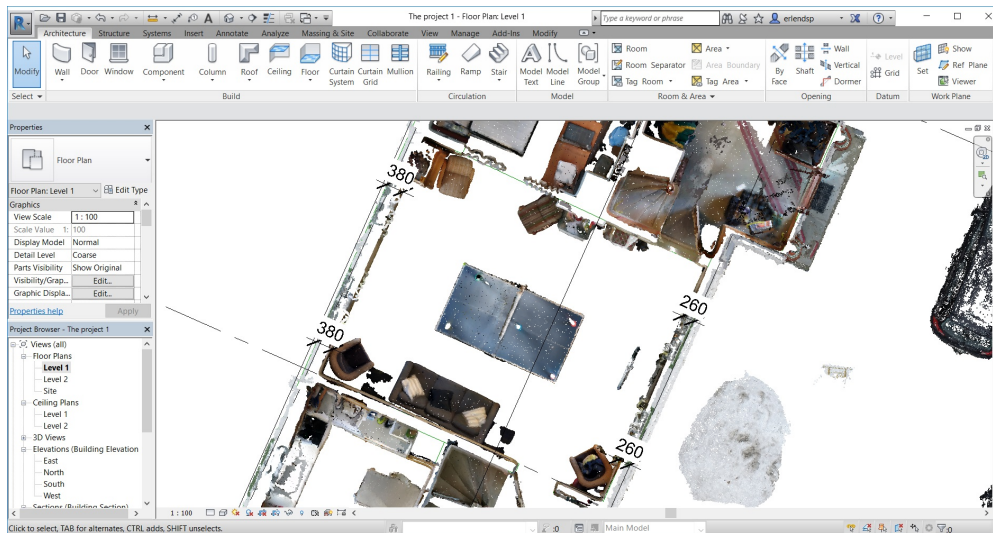
Figure 3.44: The exterior wall thickness was measured at several different points before calculating the average.

3. **Adding interior walls:** This was based on the same approach as the exterior walls. Lines were drawn, and thicknesses were measured. This is shown in Figure 3.45. To simplify the model, a generic material was used for all interior walls. The wall thickness varied across the building, hence it was necessary to create many wall-families with various thicknesses. Some walls were not orthogonal to adjacent walls, while others had variable thickness. This was probably caused by inaccuracies in the scanned model or that the farmhouse was built this way. Regardless of this, the interior walls were made by best fit by eye measurement and the result was checked against the point clouds in a 3D view afterwards.
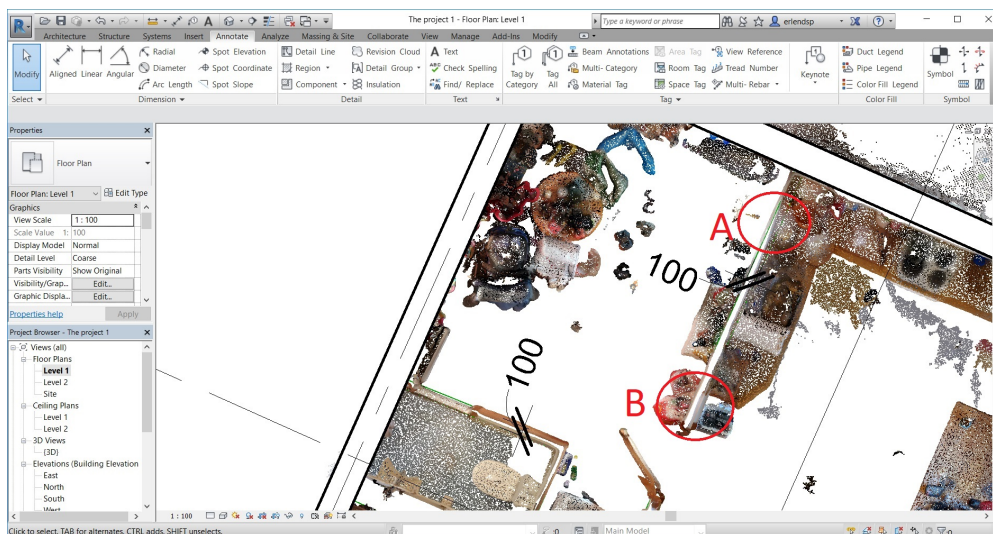


Figure 3.45: Measuring the thickness of interior walls before creating Revit families. Some walls varied in thickness. The wall is thicker in the marked circle B than in A. This is difficult to reproduce in Revit.

4. **Floors and ceilings:** Floors were made by drawing horizontal geometry and adding it
   to the appropriate level. A generic floor material was added since the floor material
   varied to much within the building. The ceilings were harder to make due to
   considerable height variation. Viewing the building from a side section was helpful
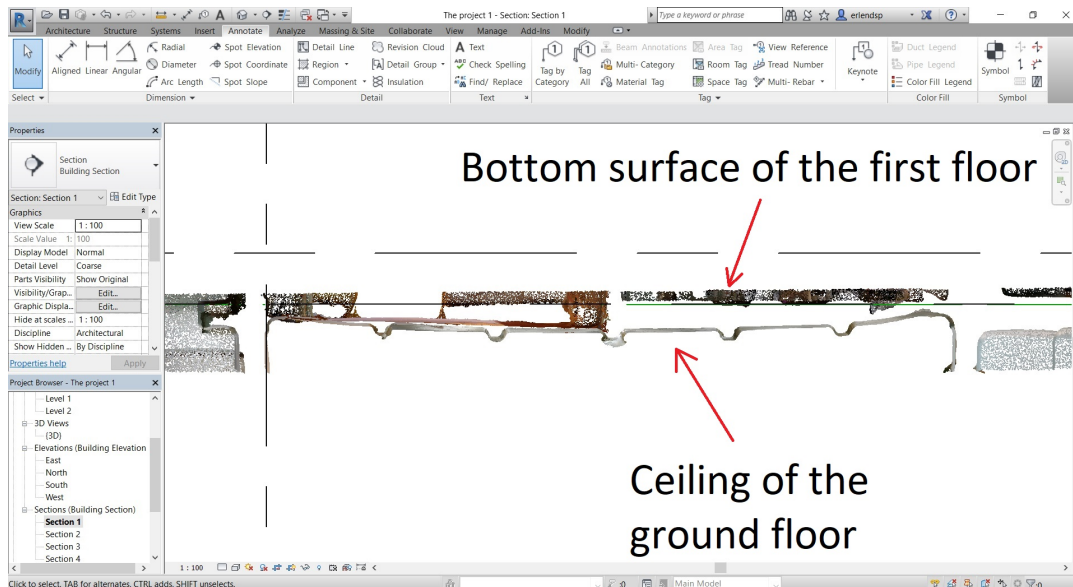   for final ceiling height adjustment.



Figure 3.46: A section of the bottom surface of the first floor and the ceiling of the ground
floor in the middle room. Some floors are sagging quite much which is hard to remake as a
BIM in Revit.

5. **Adding the roof:** The roof was created by adding reference points at the roof's top and
   bottom according to the point cloud. Revit could then automatically generate a roof
   fit for the boundary conditions chosen. Some minor adjustments had to be made to
   make it fit exactly with the point cloud.

6. **Terrain modeling:** By using the *Toposurface*-tool in Revit, while watching the point
   cloud in a 3D view, it was simple to adjust the contour lines by selecting a variation of
   altitudes. By viewing the terrain from various angles, it was possible to see where the
   BIM terrain did not fit with the point cloud. This is shown in Figure 3.47.
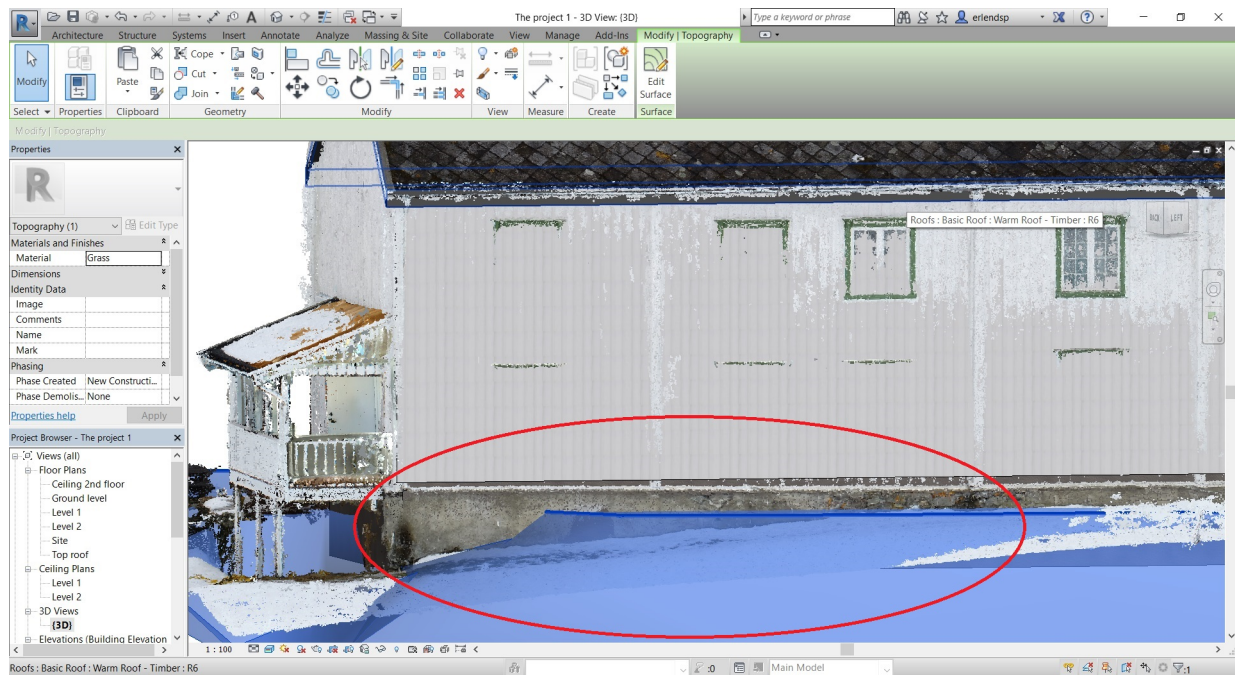
Figure 3.47: Watching the terrain around the building from various angles and adjusting the terrain to make it fit with the point cloud. The encircled area has to be adjusted.

7. **Creating stairs:** Creating the stair properly and with an exact fit to the point cloud was difficult due to lack of experience in Revit. The automatic stair creation was difficult to fit the exact geometry of the real stairs in the house. Still, the idea of creating stairs by using the point cloud geometry as a template was possible.

8. **Downloading objects from BIM database:** The standard Revit version only include a limited number of objects like doors, windows, tables and chairs. To obtain object like those in the real building, it is either possible to design them in Revit, or download similar objects from a BIM database and adjust them while modelling. The first option is time consuming and difficult, while downloading objects provides a faster, but not as accurate solution. As the goal of this thesis is to test if it is possible to design a model by using the scan as a template, and not create a 100% accurate model, objects were downloaded from online databases. They were picked from the following databases: *National BIM library, BimObject* and *RevitCity* [10][58][59]. These web sites contain thousands of objects. Picking objects for downloading is shown in Figure 3.48.

Figure 3.48: Choosing BIM object from the *BIMObject* online database [58].

9. **Adding doors:** As the doors in the building are old and unique, there were no object in
any database which looked like the real doors. Instead of downloading a vast number
of different doors, a generic door was chosen with adjustable size to fit with the
various doors in the model. Fitting the door to the right position worked out well by
using the level view as shown in Figure 3.49. After all doors had been placed by doing
this, the result was checked in a 3D view and the final adjustments were made.
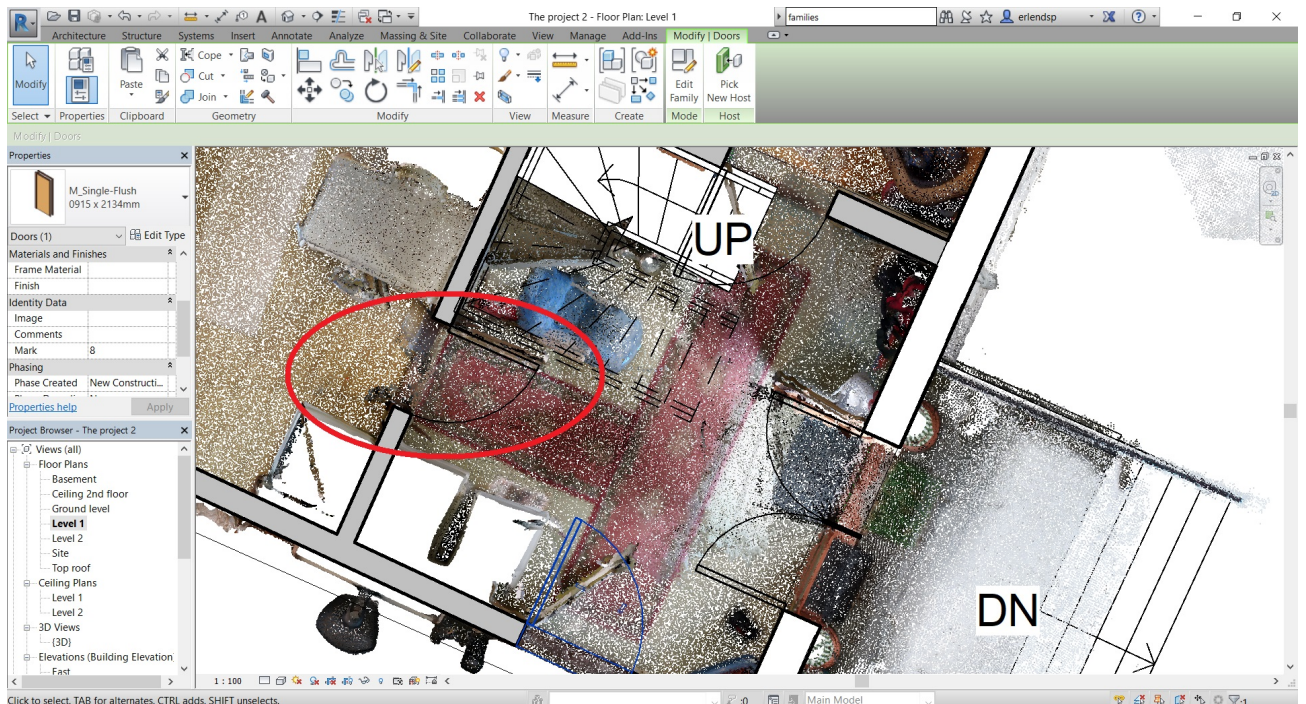


Figure 3.49: Putting doors at the right position and adjusting the size in Revit plan view.

10. **Adding windows:** Adding windows were done similar to adding doors. Model helping
    lines had to be drawn, as it was hard to find the exact position and size in the level
    view. The windows were placed into the right position before the parameters were
    adjusted to make the size as in the point cloud. The final adjustments were done in a
    3D view by watching the BIM and the point cloud in the same view. This is shown in
    Figure 3.50.
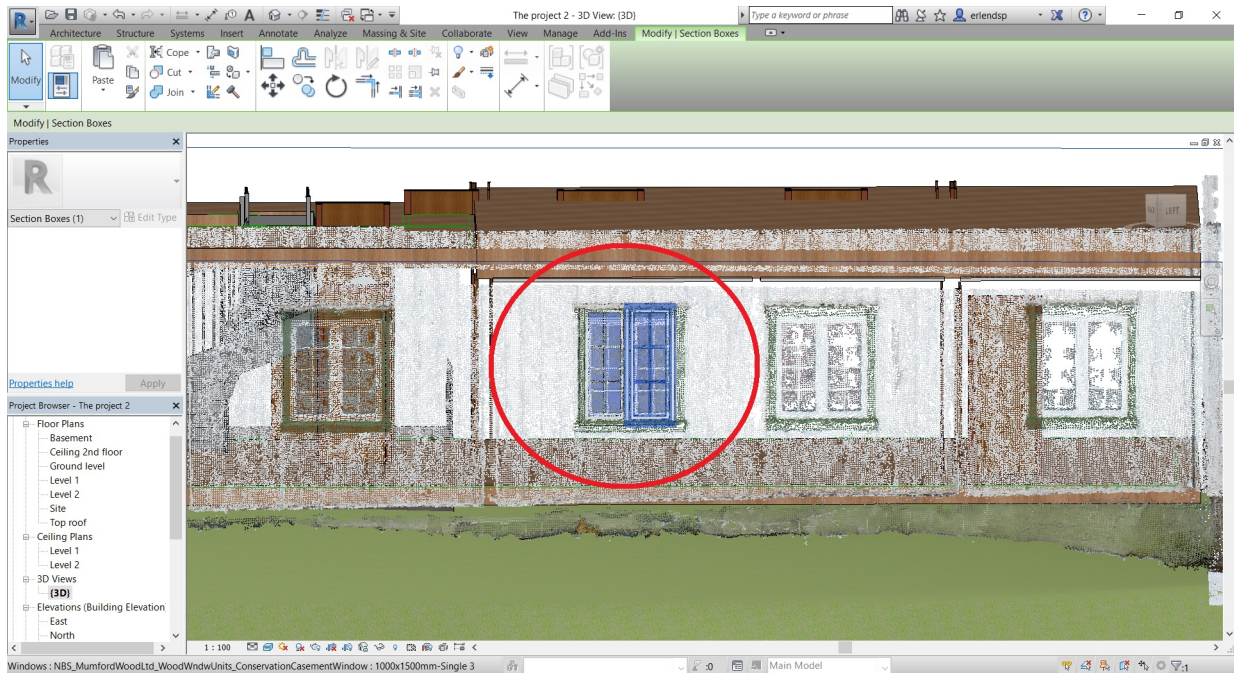


Figure 3.50: The final adjustments of windows were made in 3D view showing both the point
cloud and the BIM.

11. **Final adjustments and extrusions:** There were still some details missing in the
    model. For example, the entrance area on the northern side of the house. These
    detailed areas were made by looking at the point cloud from various angles, doing
    some measurements and adjusting the different objects according to the point cloud
    as best fit. The point cloud and the BIM of the northern entrance are shown in Figure
    3.51.
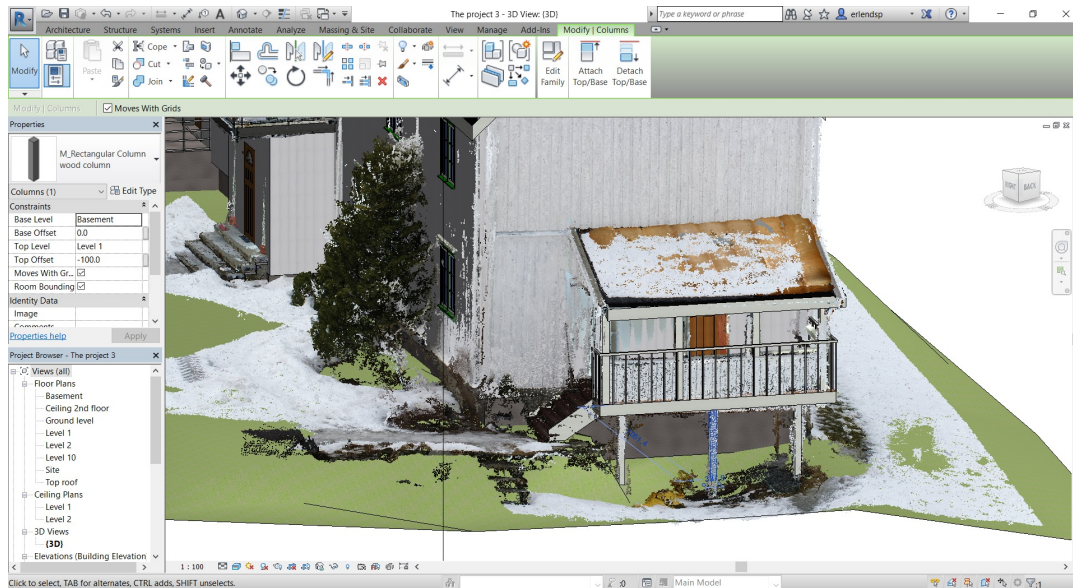
Figure 3.51: The northern exit, both BIM and point cloud in the same view.

12. **Objects:** To show how Scan to BIM can be used to add BIM objects and make rooms
    look realistic, one room was made more realistic. This was done by adding the right
    colours to floors and walls, as well as adding some objects to the room. The objects
    were downloaded from BIM databases and adjusted to fit the model. Adjustment and
    fitting of a couch is shown in Figure 3.52. Some of these objects were hard to adjust as
    the design often proved to be rigid and sometimes not adjustable at all. To make the
    objects more realistic, they would have to be created from scratch. Colours were
    added to walls and objects by taking a sample from photographs of the real objects.
    These samples were used to add colour to new materials. The objects and walls were
    further painted by using the materials created from the photographs. Editing of an
    object is shown in Figure 3.53
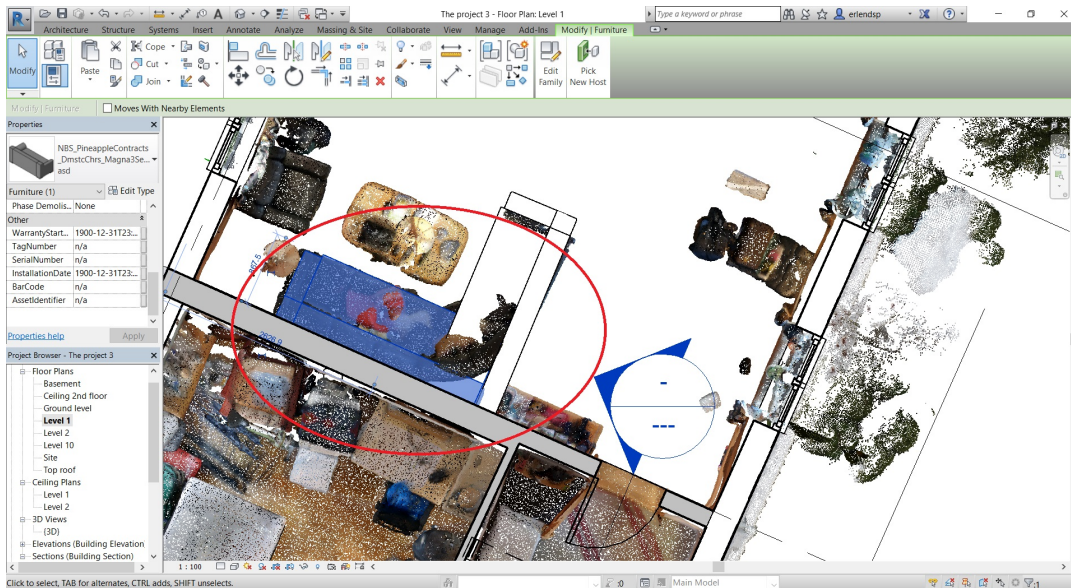
Figure 3.52: Adjusting the couch BIM object to fit the point cloud in level view.
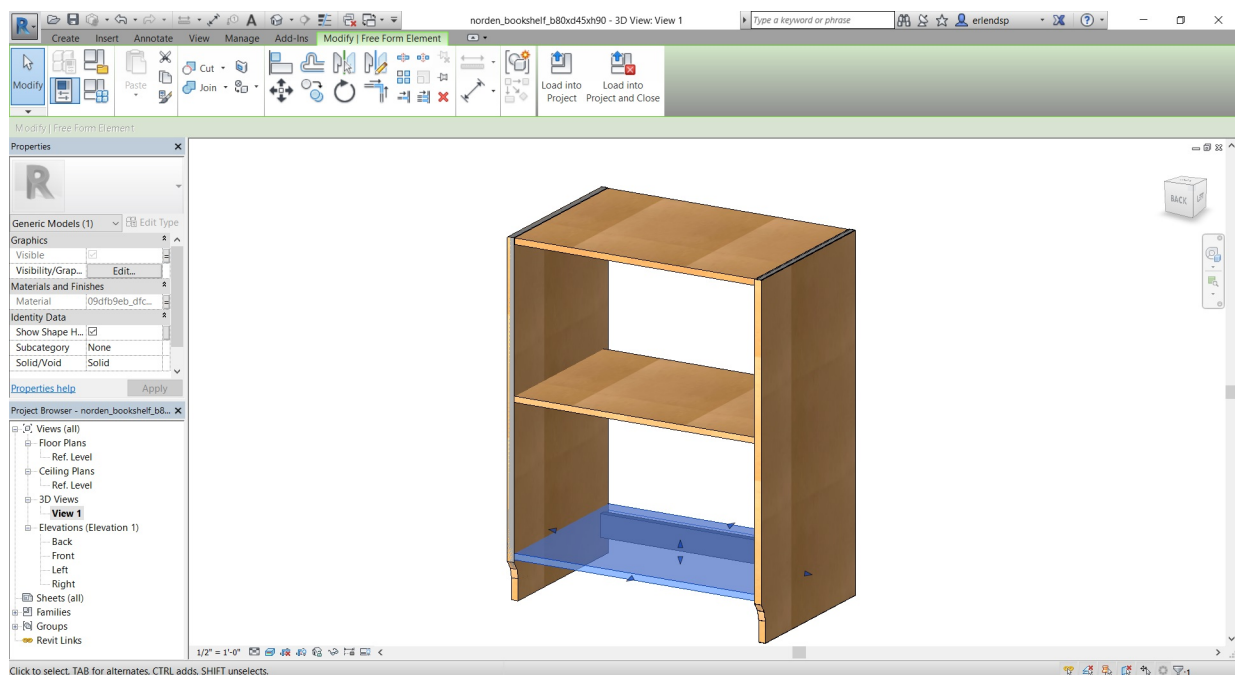


Figure 3.53: The same object as chosen from the database in Figure 3.48. Changing the parameters of the BIM to look like the real object.

At this point the BIM had all the basic geometry of the point cloud, and parts of the model also contained objects with useful information. It was possible to continue improving the model by adding more objects, better information or adding more accurate geometry. This could be done to a point where the model would look almost exactly like the real building. As this would take a vast amount of time and the goal of creating a BIM out of smartphone 3D scanning was achieved, there was not any point in creating a more detailed model for this experiment. Hence, further modelling was stopped and the results were analyzed.

# Chapter 4

# Results

## 4.1 The Model

This chapter shows the final product after processing the scan and creating the BIM. The processed scans are displayed in Autodesk ReCap, while the BIM and the BIM combined with the point cloud are displayed in Revit. As the results contain some directional explanations, Figure 4.1 displays the orientation of the farmhouse:



Figure 4.1: Orientations of the farmhouse.

### 4.1.1 Processed scanned models, made from 3D scanning:

ReCap displays the scanned model in a 3D view. Figure 4.2-4.12 shows the 3D model of the farmhouse from various views and sections:

Figure 4.2: The farmhouse viewed from the North-East.



Figure 4.3: The Farmhouse viewed from South-West.

Figure 4.4: A cut of the ground floor viewed from the West.



Figure 4.5: A cut of the first floor viewed from the West.



Figure 4.6: A section of the house displayed from the West.

Figure 4.7: A section of the farmhouse displayed from the South.



Figure 4.8: A section from the South-Eastern part of the building.

Figure 4.9: The first floor, on the southern part of the building displayed from above. When looking at the scan from such a view, it is possible to see the size of various objects: For example, parts of the chimney is placed in the middle of the red circle in this figure.



Figure 4.10: The same living room as shown in Figure 4.9, displayed from a side view.

Figure 4.11: The first floor of the Southern part viewed from above.



Figure 4.12: The middle room displayed from the South.

### 4.1.2 BIM

Revit can display the BIM in various views. These vary from 2D views, like floors and sections, to a wide range of 3D views like *Wireframe, Realistic* and *Realistic Ray Trace.* The final BIM will further be displayed in Figure 4.13-4.26 to show the model from a variation of angles as well as showing some of the potential for various applications:
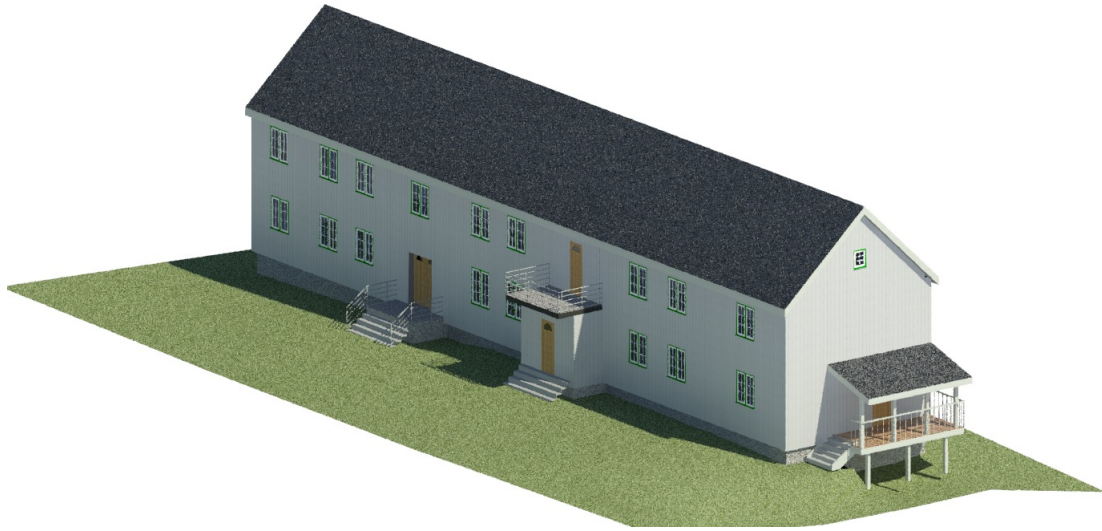


Figure 4.13: The BIM: 3D-Realistic Ray Trace view of the exterior from the North-East.
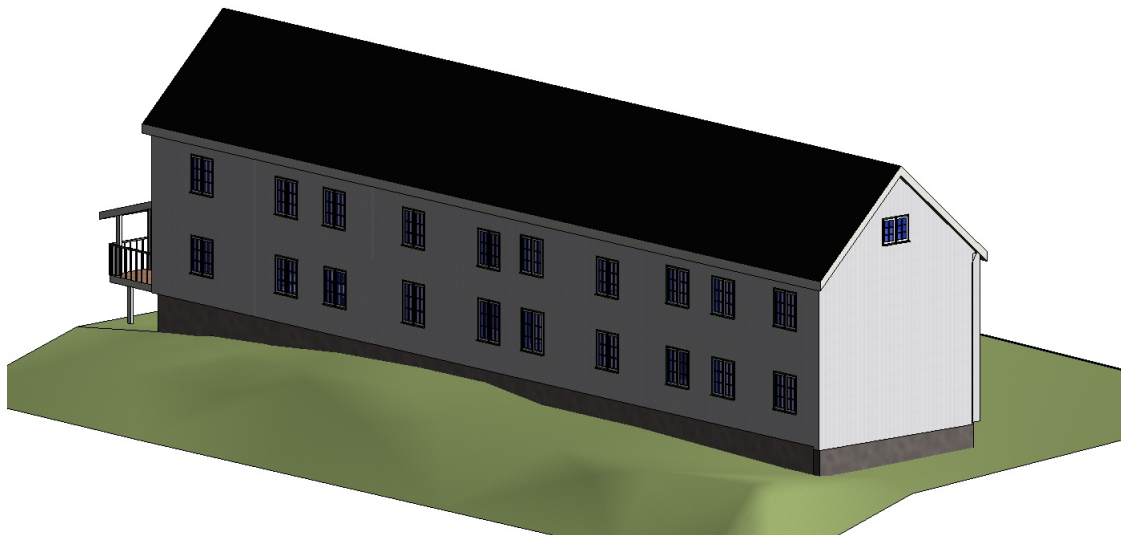


Figure 4.14: 3D Realistic view of the exterior displayed from the South-West.

Figure 4.15: 3D Realistic Ray Trace view of the ground displayed floor from the East.



Figure 4.16: 3D Realistic view of the first floor displayed from the East.



Figure 4.17: 3D Realistic view of a section of the building displayed from the West.

Figure 4.18: 3D Realistic section view of the Norther part displayed from the East.



Figure 4.19: 2D Floor view of the ground floor. Annotations of length measurements are encircled by red circles.
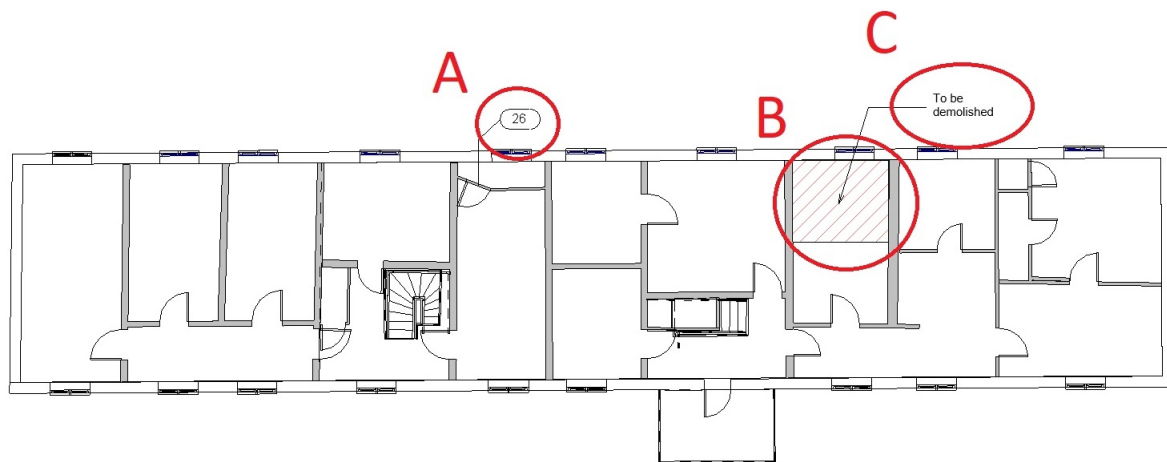
Figure 4.20: 2D Floor view of the second floor. Various possible annotations are displayed:
Circle A contains a category tag of the door. This is door number 26.
Circle B contains a marked region
Circle C contains a text explaining the marked region.
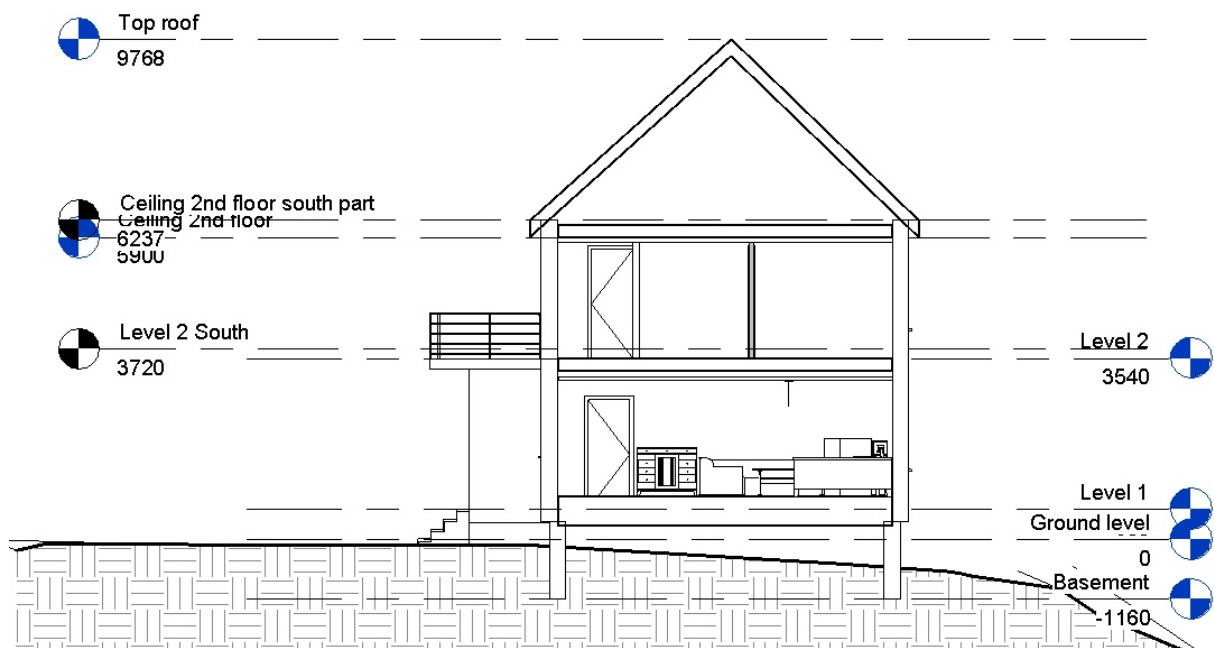


Figure 4.21: 2D Section view from the North, the section is cut around the Northern living room. Containing all levels used during BIM creation.
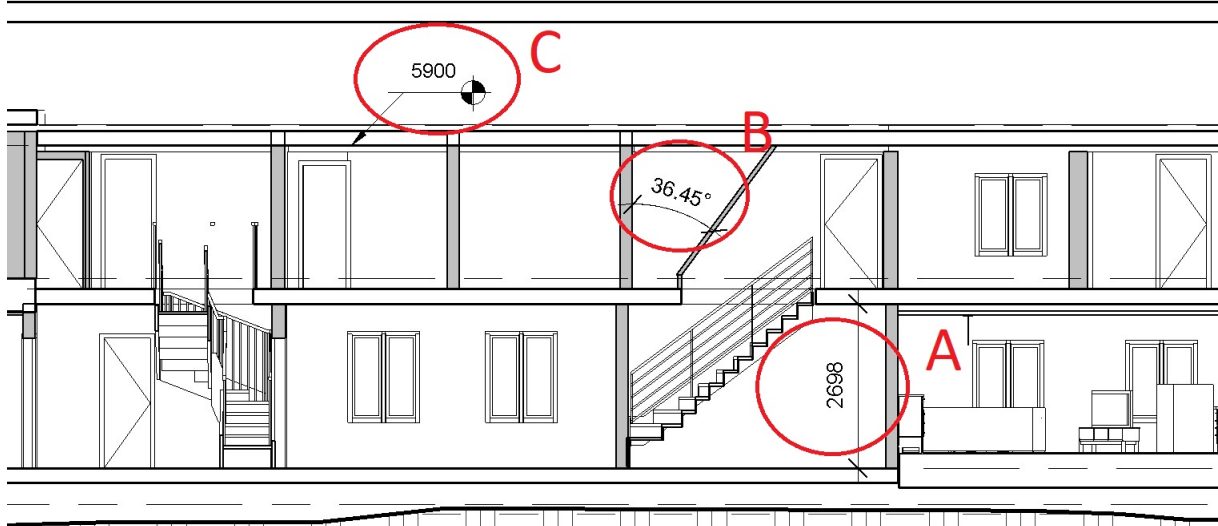
Figure 4.22: 2D Section view of the first floor. Various options for annotations are displayed:
Circle A contains a height measurement.
Circle B contains an angle measurement.
Circle C contains a description of the height above the ground for the specific point.



Figure 4.23: An example for how a design drawing sheet for this building could look like. This is including several different views and is easily printed.

Figure 4.24: 3D Realistic Ray Trace views from above. One room was made more realistic by creating some of the objects in the room as well as adding real colours to the walls, floor and furniture.



Figure 4.25: 3D Realistic Ray Trace view from the North of the Northern livingroom.
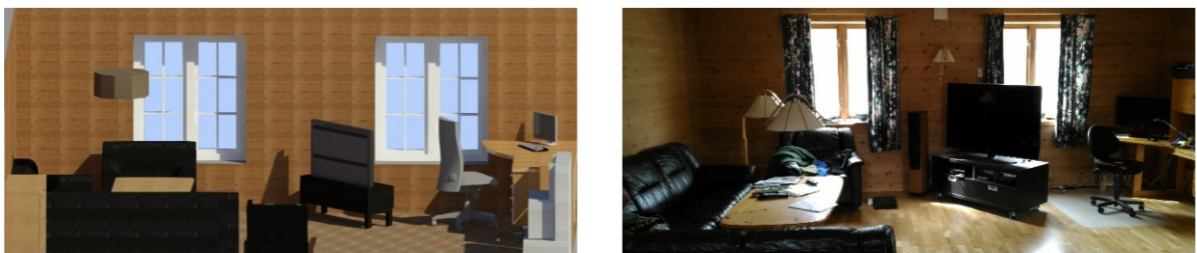


Figure 4.26: The BIM with a 3D Realistic Ray Trace view of the northern living room and a real photo of the northern livingroom.

### 4.1.3 BIM combined with point clouds

To gain an understanding about how well the point cloud fit with the BIM, it is helpful to view both the BIM and the point cloud in the same view. Figure 4.27-4.34 show both views overlayed in Revit:



Figure 4.27: BIM and point cloud in the same view: 3D Realistic view from the North-East.



Figure 4.28: 3D Realistic view displayed from the North-West.

Figure 4.29: 3D Realistic view displayed from the North.



Figure 4.30: 3D Realistic view of the first floor displayed from the East.



Figure 4.31: 3D Realistic view of the second floor displayed from the East.

Figure 4.32: A close up 3D Realistic view of the first floor displayed from the East. In this view it is possible to check that all furniture fit well with the BIM.



Figure 4.33: A close up 3D Realistic view of the Northern part of second floor viewed from above. In this view it is possible to check that all furniture fit well with the BIM.



Figure 4.34: A close up 3D Realistic view of the northern livingroom. Most BIM objects fit well with the furniture in the point cloud.

## 4.2   Efficiency

### 4.2.1   Time

During the entire experiment, the time spent on effective work was recorded. Effective work is referring only to the time spent on scanning, processing and working in the various software. Time spent planning, testing, and intervals between scannings are not included. The time spent on each part of the project is displayed in Table 4.1:

Table 4.1: Time spent

| Activity | Time spent | Comment |
| --- | --- | --- |
| All scanning | 181 min | The total time is calculated by using the numbers in Table 3.3 and 3.4 |
| All phone processing after scanning | 361 min | The total time is calculated by using the numbers in Table 3.3 and 3.4 |
| Transferring files to the computer and organizing them | 50 min | |
| Cleaning, aligning and merging in Cloud Compare | 300 min | The focus was on accuracy, meaning it could have been done much faster. |
| Final processing in ReCap | 70 min | During processing, time was spent on finding a good way of creating regions without using the select points on plane tool. |
| BIM creation in Revit | 1340 min | Lack of experience in Revit caused time delay. |
| Total time | 2302 min | |

### 4.2.2 File size

File size at various stages of the experiment is shown in Table 4.2:

Table 4.2: File size

| File | Size | Comment |
|---|---|---|
| All phone scans | 4 470 MB | Individual scans varied between 169 MB and 366 MB |
| Drone photos | 962 MB | *None* |
| Drone scan model | 300 MB | *None* |
| The model processed in Cloud Compare | 1218 MB | *None* |
| The model processed in Autodesk ReCap | 759 MB | *None* |
| Revit BIM | 44.8 MB | *None* |
| Total storage used | 7.75 GB | |

## 4.3 Accuracy

Control points A-E are on the same surfaces. The control point does not show in the BIM, which makes it impossible to measure these distances in Revit. Figure 4.35 shows where in the building the measurements were done. In the tables, "P" means *Parallel measurements,* "O" means *orthogonal measurements* and "F" mean *measurements between floors.* Table 4.3 shows the result of the measurements while Table 4.4 shows the calculated accuracy:



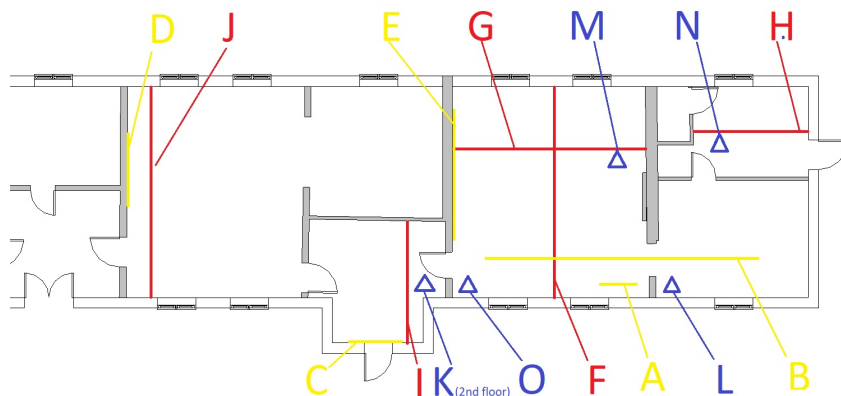Figure 4.35: An overview of where in the building the measurements were done. The ends of each line indicate the control point's positions:
The yellow lines indicate the position and length of parallel measurements.
The red lines indicate the position and length of orthogonal measurements.
The blue triangles are the positions of the control points for height measurements between floors and ceilings.

Table 4.3: Results from measurements

| Points | Real distance [m] | Distance in the Scan [m] | Distance in the BIM [m] |
|---|---|---|---|
| A1-P - A2-P | 1.220 | 1.227 | |
| B1-P - B2-P | 9.262 | 9.284 | |
| C1-P - C2-P | 1.309 | 1.323 | |
| D1-P - D2-P | 1.465 | 1.461 | |
| E1-P - E2-P | 4.370 | 4.370 | |
| F1-O - F2-O | 6.379 | 6.399 | 6.534 |
| G1-O - G2-O | 5.979 | 5.988 | 6,005 |
| H1-O - H2-O | 3.572 | 3.568 | 3.563 |
| I1-O - I2-O | 3.622 | 3.615 | 3.774 |
| J1-O - J2-O | 6.522 | 6.522 | 6.519 |
| K1-F - K2-F | 2.430 | 2.475 | 2.360 |
| L1-F - L2-F | 2.409 | 2.441 | 2.278 |
| M1-F - M2-F | 2.440 | 2.456 | 2.278 |
| N1-F - N2-F | 2.510 | 2.510 | 2.278 |
| O1-F - O2-F | 2.411 | 2.496 | 2.278 |

Table 4.4: Accuracy of measurements

| Points | Real distance [mm] | Scan error in [mm] | Scan error in [%] | BIM error in [mm] | BIM error in [%] |
|---|---|---|---|---|---|
| A1-P - A2-P | 1220 | 7 | 0.57 | | |
| B1-P - B2-P | 9262 | 22 | 0.24 | | |
| C1-P - C2-P | 1309 | 14 | 1.07 | | |
| D1-P - D2-P | 1465 | 4 | 0.27 | | |
| E1-P- E2-P | 4370 | 0 | 0.00 | | |
| Average P | | 9.4 | 0.43 | | |
| F1-O - F2-O | 6379 | 20 | 0.31 | 155 | 2.43 |
| G1-O - G2-O | 5979 | 9 | 0.15 | 26 | 0.43 |
| H1-O - H2-O | 3572 | 4 | 0.11 | 9 | 0.25 |
| I1-O - I2-O | 3622 | 7 | 0.19 | 152 | 4.20 |
| J1-O - J2-O | 6522 | 0 | 0.00 | 3 | 0.05 |
| Average O | | 8 | 0.15 | 69 | 1.47 |
| K1-F - K2-F | 2430 | 45 | 1.85 | 70 | 2.88 |
| L1-F - L2-F | 2409 | 32 | 1.33 | 131 | 5.44 |
| M1-F - M2-F | 2440 | 16 | 0.65 | 162 | 6.64 |
| N1-F - N2-F | 2510 | 0 | 0.00 | 232 | 9.24 |
| O1-F - O2-F | 2411 | 85 | 3.53 | 133 | 5.52 |
| Average F | | 35.6 | 1.47 | 145.6 | 5.94 |
| Total average | | 17.67 | 0.68 | 107.3 | 3.71 |

# Chapter 5

# Discussion

The main objective of this experiment was to determine if it is possible to create a BIM of an existing building using smartphone 3D scanning. Co-objectives, as stated in section 1.2 *Objectives*, include several aspects of smartphone Scan to BIM and the usability of this technology. The data yielded by this experiment provide convincing evidence that it is possible to create a BIM from smartphone scanning. Still, the credibility, usability and workflow of the experiment will be elaborated in this chapter. The rest of the discussion will focus on the key areas of the objectives and the results

## 5.1 Procedure

The nature of this thesi's objectives allowed several approaches to the problem. By starting the work with researching and testing several alternative approaches, this report has a solid foundation for claiming that the procedures used are well thought through. Still, more approaches could have been tested in a larger experiment to find alternative ways of performing scanning and post processing.

The data gathered in the research leading to the experiment, indicated that there are two main methods of 3D scanning using a smartphone:

- Using the smartphone's cameras to create a 3D model by photogrammetry.

- Google Tango technologies.

Based on the testing performed prior to the experiment, the photogrammetry method faced several issues when scanning the interior of the house and also result in lower accuracy. Hence, making Google Tango a better choice for this kind of scanning.

During processing, several software packages were used. Transferring the file formats from *plh/obj* to *rcp* was one of the main reasons for using several software packages. At the same

time, using several software packages provided the option of merging, cleaning and other point cloud manipulation. As a rebuttal to this point, it could be argued that it would have been better to use a single software for all steps of processing. This would make the processing easier and more user friendly. Such a software, would preferably also include BIM creation options like in Revit. Based on technology currently under development, like the EdgeWise software elaborated in section 2.3.2 *BIM preparation software*, a future software could also include automated BIM generation or object recognition. Presently, this kind of software does not exist, making it necessary to use several software packages. The chosen method proved to be simple and contained as few steps as possible, making it a feasible method for processing data from smartphone scanning.

One of the co-objective was to determine if smartphone scanning could be combined with other scanning techniques. This was tested by combining a drone photogrammetry scan with the smartphone scanning. This worked out well, and the two scans were effortlessly combined into a single 3D model. The result of the processed drone scan was a point cloud, similar to those produced by a laser scanner or other scanning equipment. This gives a good indication that it is possible to combine most scanning equipment with the current smartphone scanning techniques.

The BIM creation process was time consuming and difficult. Partly due to lack of Revit skills and experience, but also software and point cloud issues. It may seem like Revit focuses mainly on the design of new buildings, and not reproduction of existing ones. The software is especially difficult to use for designing "special" features like a wall with varying thickness or unusual objects. This made it hard to accurately reproduce parts of the building. Still, by doing the procedure described in section 3.3 *Execution*, the building was reproduced. During BIM creation, there was a constant consideration about the level of detail in the final BIM. A detailed model would further implicate that BIM creation from smartphone scanning is possible. However, the results from the BIM made in this experiment provides sufficient evidence that it is possible. Hence, a more detailed model, for example by adding chimneys or more objects as shown in Figure 5.1 would only prolong the experiment unnecessarily.

Figure 5.1: Viewing the BIM and the point cloud in the same view. Examples of things that could have been added to the BIM:
Chimneys could have been added in circle A and B.
Several objects could have been added in circle C.

## 5.2 The Model

The final BIM consist mainly of floors, walls, roof, terrain, doors, windows and stairs. Interior objects were also added in one of the rooms of the building. All these model parts are represented as BIM objects in Revit and contain various attributes. The geometry of the objects can be adjusted to fit with the point cloud, making it easy to recreate the correct geometry. Non-geometric attributes are harder to reproduce correctly in Revit. Main information added to this model is colour, material and connections to adjacent objects. To make this a more informative BIM, more information should have been implemented in the model. This is hard to do when the only reproduction basis is the point cloud from the smartphone scan.

From the figures in Section 4.1.3 *BIM combined with pointclouds*, it can bee seen that the BIM is tightly fit with the reality captured by the scan. Some parts of the model are more correctly recreated than others. Looking at Figure 5.2, which shows the ground floor of the building, there are clear indications that some parts are more accurately represented than others. As framed in Figure 5.2, the visible floor varies depending on the room. This mean that the BIM do not reproduce all floors accurately. A solution to this issue could have been to create more Revit Levels and add various heights to the floors on the same level. This was done on the second floor because there was a large deviation of height, but it was skipped on the ground floor. On the contrary, by looking at the windows in Figure 5.2, it looks like there are no deviation between the point cloud and the BIM. This provides confirmatory evidence that the windows are correctly reproduced in the BIM. The same applies to doors, objects in the point cloud and the walls. This indicates that the final BIM is a good reproduction of
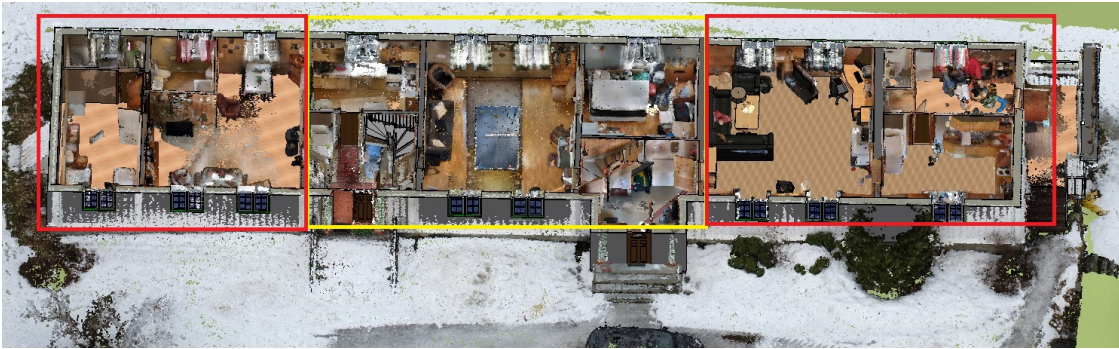
reality.



Figure 5.2: When viewing the BIM and the point cloud in the same view, some parts of the building has visible BIM-floor while other have visible point cloud floors. This happens because the BIM do not correctly recreate the varying height of the floors.
The BIM-floor is visible inside the red squares.
The point cloud floor is visible inside the yellow square.

## 5.3   Efficiency

The effective time spent scanning and processing totals about 9 hours, 3 hours scanning and 6 hours processing. If this was to be done in another construction project, there would have been some time spent on planning and preparing for scanning as this was a quite complex building containing a lot of various objects. Scanning a less complex building would also shorten the time spent scanning and processing further. This means that the scanning itself could have been done within a relatively short amount of time, especially if the scanning quality was not important. The same applies to all processing before BIM creation. In this experiment, the processing proceeded slowly, due to lack of processing experience and a desire for high accuracy. Still, only about 6 hours were spent on post processing the scanned data on the computer. The rest of the time, about 22 hours, was spent on creating a BIM from the scanned data. Even when the BIM contained only the "basics" and could have been worked on for dozens of more hours, the time spent on BIM creation is more time than both scanning and all processing combined.

Based on the evidence on scanning and processing time for a first-time project like this, it seems fair to suggest that time spent scanning and processing could have been drastically reduced for a second project. As for BIM creation, it would also be faster the second time. Still, a more detailed BIM for reproduction of the entire building would prolong the BIM creation time further.

When looking at efficiency of using the technology, the current debate revolves around comparing smartphone scanning for BIM creation to commercial methods with the same result. The commercial alternatives would include scanning the building using a laser scanner or

measuring the distances manually and then creating a BIM based on those results. All alternatives would include BIM creation, so most of the time would probably be spent doing this on the other alternatives as well. Testing the time spent on these methods is beyond the scope of this thesis, but a hypothetical comparison would be:

- Creating the BIM based on a laser scanner would involve many of the same steps as by using a smartphone. Scanning these rooms using a laser scanner would be troublesome as many of the rooms are small and contain many obstacles. This would call for a need to scan the rooms from several various locations in each room, making it fair to assume that using a laser scanner would be slightly more time consuming than using a smartphone scanner.

- By using manual measurements, there would not be any need for processing the scan. Still, much time would be spent on measuring all distances in the building, not only the height of all floors and distance between walls, but also the position and size of windows, doors, stairs and so on. BIM creation would also be more troublesome as there would not be a point cloud to use as a template while designing. Based on time spent on measuring the control points and doing BIM creation, it is fair to assume that manual measurements for BIM creation would be more time consuming than using smartphone scanning.

## 5.4 Accuracy

The data gathered in the experiment appears to suggest that the scanning is surprisingly accurate, while BIM reproduction is prone to error. The total average error on the scan is 18 mm or 0.68%, compared to 107 mm or 3.71% in the BIM. Table 4.4 indicates that the error in the distances between floors (F), is several times the error of parallel and orthogonal measurements in the scanned file and more than twice the error in the BIM. This applies for all measurements between floors, making it reasonable to believe that they are caused by the same error. Looking at merely the parallel and orthogonal measurements, the average error is 8.7 mm or 0.29%. Wich is quite accurate. Using only orthogonal measurements for the BIM gives 69 mm or 1.47% error, which is a small, but not neglectable error.

The scanning accuracy indicate that scanning by smartphone could provide a usable result for several purposes, with accuracy almost the same as a laser scanner. As the smartphone scanner appears to be more feasible for scanning small and medium rooms than a laser scanner, the smartphone scanner has the potential of substituting the laser scanner as the main tool used for scanning.

As the scanning accuracy was high, it is reasonable to believe that the BIM error is due to inaccuracies during BIM creation. The reason is probably that major parts of the BIM was

created on a best fit by visual judgement. By establishing a better procedure or software for BIM creation, it is likely that the BIM error would be reduced drastically.

**So how much can the results from the accuracy experiment be trusted?**

15 control points were used for measuring the accuracy in this experiment. Five of these could not be used for BIM accuracy measurements, making it a total of 10 point to measure the discrepancies between reality and the final BIM. That is too few measurements to obtain a conclusive statistic of the accuracy. As the point of measuring the error was to gain an understanding of the model's accuracy and not make a conclusive statistic, it could be argued that the number of control points are sufficient.

Several parts of the accuracy measurement are prone to error:

- The manual measurement using the laser distance measurer are prone to error both regards to placement of the control points and by the execution.

- Clicking the pointer at the exact right position on the control point, provides a source of error on the MeshLab measurements.

- There were several sources of error during BIM creation.

By combining all these factors of error, there are reasons to believe that there should be a variation of the accuracy results. As this is a critical question with regards to further usability of the technology, it would be interesting to perform an experiment for measuring accuracy of smartphone scanning using more control points in the future.

## 5.5   Usability

The results provide convincing evidence that Scan to BIM using smartphone for 3D scanning is a feasible tool. There are, however, some issues related to accuracy and efficiency. Based on the accuracy measured on the BIM, the technology is currently not suitable for creating detailed drawings, but could possibly be used as a basis for redesign or other BIM applications like documentation, demolition and structural analysis.

A further comparison of smartphone scanning with other scanning techniques is listed in Table 5.1:

Table 5.1: Comparing smartphone scanning to current scanning methods

|  | **Pros compared to using smartphone scanning** | **Cons compared to smartphone scanning** |
|---|---|---|
| **Laser scanning** | Laser scanners provide millimeter precision, making it slightly more accurate than a smartphone. The software dedicated to the specific laser scanner would likely make processing simpler and faster. | The small size and flexible positioning of the smartphone make it suitable for accessing small and/or cramped areas, as shown in Figure 5.3. The smartphone is superior in such areas. The scanning is probably faster using a smartphone than a laser scanner. |
| **Manual measurements** | Manual measurements require neither scanning equipment nor software, except for a BIM package where data is entered manually. | Using a smartphone scan would give more detail to the recreation. BIM creation would be faster as the point cloud is used as a template when recreating. |



Figure 5.3: By using a phone instead of a laser scanner, it is possible to access more areas of the building from several angles.

Table 5.1, provides some evidence that smartphone 3D scanning for BIM creation has the potential of being superior to current methods of making BIMs of existing buildings.

Based on the experience gained from this experiment, it seem to be compelling evidence that this technology has the potential of being used in areas of application as encircled in Figure 5.4:
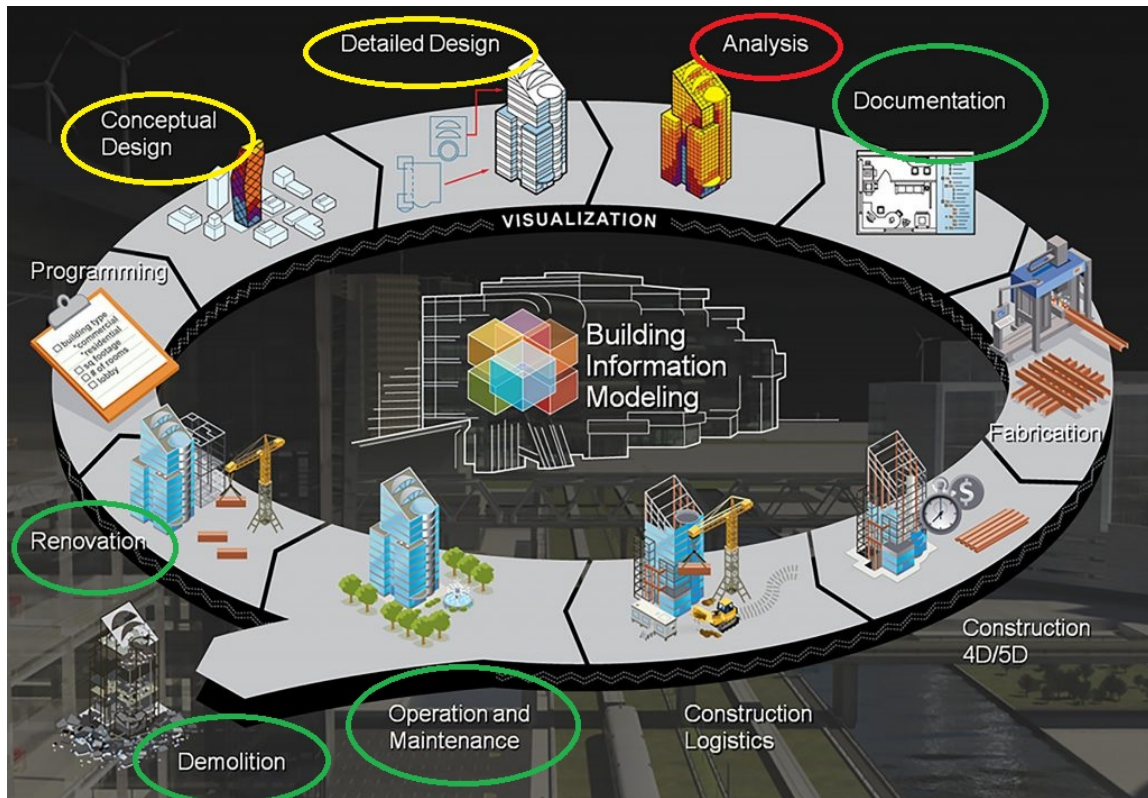


Figure 5.4: Showing potential applications of smartphone scan to BIM out of the current BIM applications if the technology is further developed:
Green circles show likely areas of application.
Yellow circles show possible areas of application.
Red circles show unlikely, but no impossible areas of application. [17].

As the technology provides a new, efficient way of creating a 3D model or a BIM of an existing building or environment, it is possible to utilize it in other areas than regular BIM creation for construction. Such areas could include:

- Scanning and creating a 3D model of a building to be used for visualization in the real-estate business.

- Military recognizance. Making it possible to scan and print a 3D model of a building or an area to be used for planning operations.

- Visualization and documentation in other industries like shipping and manufacturing. For example, a ship without its design drawings could be scanned, and the 3D model further used for documentational purposes.

# Chapter 6

# Conclusion

The experiment conducted in this Master's Thesis proves that it is possible to create a BIM from smartphone 3D scanning. It also proves that it is possible and effortless to combine a smartphone scan with other scanning techniques. By measuring efficiency and precision of scanning and BIM creation, there are convincing evidence that smartphone scanning has the potential of becoming a useful tool for the construction industry and other applications. The accuracy of scanning was surprisingly high, which opens for a vast amount of applications. BIM creation proved slightly error prone, as the techniques of turning the scan into a BIM is still mainly manual and based on best fit by eye measurement. By establishing another method for BIM creation from 3D scanning, there are immense potential of this technology both for scanning and BIM creation.

## 6.1   Recipe for Smartphone Scan to BIM Project

Based on the experience gained from this experiment, the following procedure is suggested for future Scan to BIM projects using smartphone 3D scanning:

1. Create merging cones like shown in Figure 3.21

2. Plan the scan and prepare the building by opening doors and placing merging cones.

3. Scan the building in a stepwise fashion. Preferably each scan should not take more than 10-15 minutes.

4. Complete phone processing and upload the files to a Google drive or similar storage.

5. Take photos of parts of the building to reproduce the same colours in BIM creation.

6. If needed, scan distant areas of the building like the roof or exterior parts using drone photogrammetry.

7. Open the files in Cloud Compare.

(a) Use the *Convert texture/material to RGB*-tool for each scan.

(b) Sample each scan by using the *Sample points*-tool and chose point cloud density for sampling.

(c) Align all point clouds using the *Align two clouds*-tool.

(d) Clean the point clouds using the *Cross section*-tool.

(e) Merge the scan using the *Merge multiple clouds*-tool.

(f) Export the file as a *E57* file format.

8. Open the *E57* file in Autodesk Recap and check the model for errors.

9. Divide the model into regions like walls and floors, using either the plane tool or section box for choosing the points.

10. Export the model as a *rcp* file.

11. Open a new project in Revit and import the point cloud. For the rest of the BIM creation, hide the point cloud regions not in use.

12. Create views, level and sections, using best fit by eye measurement.

13. Design floors, walls and terrain by using the point cloud as a template. It is necessary to create Revit families. When doing this, change the materials to fit with the photos taken in step 5 and create one family for each wall/floor thickness.

14. Design doors, windows, stairs etc by downloading or creating the appropriate Revit families and adjusting the scale to fit the point cloud. Add objects to the model by inspecting the point cloud in a 3D view and adjust the position/size.

15. Alternatively add objects or details by downloading or creating new objects in Revit.

Figure 6.1 displays the suggested workflow.



Figure 6.1: The software workflow and file formats

## 6.2 Further Work and Improvements

During the experiment, areas of improvements and possibilities for further work were discovered:

- The Constructor app was unstable and crashed several times while scanning. This made scanning troublesome and time consuming. It also made scanning large areas in a single scan hard, as the app crashed or the phone was unable to process the scan. Hence, a further development of the app to make it more stable would drastically improve scanning quality.

- A single software specified for receiving smartphone scans, processing these and making them ready for BIM creation would simplify the post processing.

- Other methods of BIM creation based on a scanned point cloud could be tested.

- An automated BIM generation software from point cloud geometry would make BIM creation faster and more accurate.

- As the scanning accuracy is essential for understanding future applications of smartphone scanning. A detailed experiment to discover the scanning accuracy would be useful.

# Bibliography

[1] Erlend Spets. *"Scan to BIM" - A brief introduction, Project assignment.* Department of structural engineering, NTNU, Trondheim, Norway, December 2017.

[2] Mostafa A-B Ebrahim. *3D Laser scanners: History, applications, and future, Review.* King Abdulaziz University, Jeddah, Saudi Arabia, October 2014. https://www.researchgate.net/publication/267037683_3D_LASER_SCANNERS_HISTORY_APPLICATIONS_AND_FUTURE [March 4, 2018].

[3] MachineDesign. Basics of rotary encoders: Overview and new technologies. http://www.machinedesign.com/sensors/basics-rotary-encoders-overview-and-new-tech [May 11, 2018].

[4] LiDAR-UK. How does lidar work? http://www.lidar-uk.com/how-lidar-works/ [March 04, 2018].

[5] RP PhotonicsEncyclopedia. Phase shift method for distance measurements. https://www.rp-photonics.com/phase_shift_method_for_distance_measurements.html [March 01, 2018].

[6] Gamini Dissanayake, Salah Sukkarieh, Eduardo Nebot, and Hugh Durrant-Whyte. The aiding of a low-cost strapdown inertial. *EEE Transactions on Robotics and Automation*, 17:731–747, October 2001. https://ieeexplore.ieee.org/abstract/document/964672/ [June 2, 2018].

[7] P. Fiadeiro. *Calculation of exact position using intersection, resection and distances with least squares adjustment.* Internatinal Hydrographic Review, Monaco, January 1987. https://journals.lib.unb.ca/index.php/ihr/article/viewFile/23406/27181 [March 4, 2018].

[8] Henri P.Gavin. *The Levenberg-Marquardt method for nonlinear least squares curvefitting problems.* Department of Civil and Environmental Engineering, Duke University, North-Carolina, United States, March 2017. http://people.duke.edu/~hpgavin/ce281/lm.pdf [March 4, 2018].

[9] Rebakka Volk, Julian Stengel, and Frank Schultmann. Building information modeling (BIM) for existing buildings — literature review and future needs. *Automation in*

*Construction*, 38:109–127, March 2014.
https://ac.els-cdn.com/S092658051300191X/1-s2.0-S092658051300191X-main.pdf?_tid=spdf-e07d520f-5062-4d36-9fe2-b2251cbbe35f&acdnat=1518786527_014b1c915a4952a40b0aa2f089186c1c [Feb 16, 2018].

[10] NBS National BIM libary. What is a BIM object?
https://www.nationalbimlibrary.com/about-bim-objects [Feb 28, 2018].

[11] Chuck Eastman, Paul Teicholz, Refael Sacks, and Kathleen Liston. *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors 2nd edition*. Wiley, August 2008.
http://staff.www.ltu.se/~marsan/P7006B/Literature/Eastman%20et%20al%20-%20BIM%20handbook/ch2.pdf [Feb 16, 2018].

[12] BRANZ. Productivity benefits of bim.
http://www.mbie.govt.nz/about/whats-happening/news/document-image-library/nz-bim-productivity-benefits.pdf [March 01, 2018].

[13] Zoubin Ghahramani. Bayesian modelling.
http://mlg.eng.cam.ac.uk/zoubin/talks/lect1bayes.pdf [June 9, 2018].

[14] Towards Data Science. A zero-math introduction to markov chain monte carlo methods.
https://towardsdatascience.com/a-zero-math-introduction-to-markov-chain-monte-car [June 4, 2018].

[15] Viorica Pătrăucean, Iro Armeni, Mohammad Nahangi, Jamie Yeung, Ioannis Brilakis, and Carl Haas. State of research in automatic as-built modelling. *Advanced Engineering Informatics*, 29:162–171, April 2015.
https://www.sciencedirect.com/science/article/pii/S1474034615000026 [March 7, 2018].

[16] Xuehan Xiong, Antonio Adan, Burcu Akinci, and Daniel Huber. Automatic creation of semantically rich 3d building models from laser scanner data. *Automation in Construction*, 31:325–337, January 2013.
https://ac.els-cdn.com/S0926580512001732/1-s2.0-S0926580512001732-main.pdf?_tid=63bca942-6533-4b8a-8557-16e47f59e13b&acdnat=1520415761_518a8f88da1b5bdaa715b7c37f97d49f [March 7, 2018].

[17] Information quality. Iq building information modelling (bim) services.
https://www.iq-im.com/iq-announce-bim-service-offering/ [March 07, 2018].

[18] V.Blanz, A.Mehl, T. Better, and H.-P Seidel. *A Statistical Method for Robust 3D Surface Reconstruction from Sparse Data*. Max-Planck-Institut für Informatik, Saarbruck,

Germany, Department of Restorative Dentistry, Ludwig Maximilians University, Munchen, Germany and University of Basel, Departement Informatik, Basel, Switzerland, September 2004.
https://pdfs.semanticscholar.org/4bf6/6c032711cea3bad46dcdf62d535fb6ae04ad.pdf [March 8, 2018].

[19] Vladislav Kraevoy and Alle Sheffer. *Cross-Parameterization and Compatible Remeshing of 3D Model*. Department of Computer Science University of British Columbia, Vancouver, Canada, August 2004.
https://www.cs.ubc.ca/~vlady/cross/X-siggraph.pdf [March 8, 2018].

[20] Wikipedia. Polygon mesh.
https://en.wikipedia.org/wiki/Polygon_mesh [March 08, 2018].

[21] Colin Smith. *On Vertex-Vertex Systems and Their Use in Geometric and Biological Modelling*. Department of computer science, University of Calgary, Alberta, Canada, Aprl, 2006.
http://algorithmicbotany.org/papers/smithco.dis2006.pdf [March 8, 2018].

[22] Piamulholland. Mesh construction.
https://piamulholland.wordpress.com/unit-66/mesh-constrution/ [March 08, 2018].

[23] Istituto di Scienza e Tecnologie dell'Informazione "A. Faedo. Software.
http://www.isti.cnr.it/research/software.php [March 15, 2018].

[24] MeshLab. Features.
http://www.meshlab.net/ [March 15, 2018].

[25] Cloud Compare. Cloud compare wiki.
http://www.cloudcompare.org/doc/wiki/index.php?title=Introduction#Some_technical_considerations [March 15, 2018].

[26] Autodesk. Recap.
https://www.autodesk.com/products/recap/overview [March 19, 2018].

[27] BIMtek. ClearEdge3d - EdgeWise Structure.
http://bimtek.com.au/new-blog/2017/11/27/clearedge3d-edgewise-structure [March 19, 2018].

[28] ClearEdge3D. EdgeWise building modelling tools.
http://www.clearedge3d.com/products/edgewise-building/ [March 19, 2018].

[29] Quora. What is the most popular software for bim architectural modeling?
https://www.quora.com/What-is-the-most-popular-software-for-BIM-architectural-mod
[April 24, 2018].

[30] Wikipedia. Autodesk Revit.
     https://en.wikipedia.org/wiki/Autodesk_Revit [April 24, 2018].

[31] Autodesk. Revit.
     https://www.autodesk.co.uk/products/revit/overview [April 24, 2018].

[32] Malavida. Revit architecture.
     http://www.malavida.com/fr/soft/revit-architecture/#gref [April 24, 2018].

[33] RevitCity. Objects.
     https://www.revitcity.com/downloads.php [April 24, 2018].

[34] Autodesk. Robot Structural Analysis Professional.
     https://www.autodesk.com/products/robot-structural-analysis/overview
     [April 24, 2018].

[35] Autodesk. Dynamo studio.
     https://www.autodesk.com/products/dynamo-studio/overview [April 24, 2018].

[36] Canvas. The new power tool for home improvement.
     https://canvas.io/pricing [April 26, 2018].

[37] Wikipedia. Tango (platform).
     https://en.wikipedia.org/wiki/Tango_(platform) [April 26, 2018].

[38] Youtube. Project tango - journey.
     https://www.youtube.com/watch?time_continue=210&v=44vppay5UDc [April 26,
     2018].

[39] Youtube. What's new on tango (google i/o '17).
     https://www.youtube.com/watch?v=BOrg2oc3-rQ [April 26, 2018].

[40] ARCore Overview. ARCore overview.
     https://developers.google.com/ar/discover/ [April 26, 2018].

[41] Pocket-lint. Google tango explained: What could it do and why was it shut down?
     https://www.pocket-lint.com/phones/news/google/137842-google-tango-explained-what
     [April 26, 2018].

[42] Abdoulaye A. Diakite and Sisi Zlatonova. First experiments with the tango tablet for
     indoor scanning. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial
     Information Sciences*, III-4, July 2016.
     https://repository.tudelft.nl/islandora/object/uuid:
     05aee2fa-02e3-4590-9d35-76ff1fc3cbb7/datastream/OBJ [April 29, 2018].

[43] Vladyslav Usenko, Jakob Engel, Jorg Stuckler, and Daniel Cremers. *Direct Visual-Inertial Odometry with Stereo Cameras.* 2016 IEEE International Conference on Robotics and Automation (ICRA), August 2016.
https://vision.in.tum.de/_media/spezial/bib/usenko16icra.pdf [April 29, 2018].

[44] Tango Project. Motion tracking.
https://tangoprojectweb.wordpress.com/overview-about-tango-project/4/ [April 29, 2018].

[45] GitHub. Coordinate system confusion.
https://github.com/ethz-asl/ethzasl_sensor_fusion/issues/20 [April 29, 2018].

[46] Sebastian Thrun Yufeng Liu, Zoubin Ghahramani Daphne Koller Andrew Y. Ng, and Hugh Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *Sage journals issue 7-8*, 23:693–716, August 2004.
http://journals.sagepub.com/doi/pdf/10.1177/0278364904045479 [April 29, 2018].

[47] Simon Lynen, Torsten Sattler, Michael Bosse, Joel Hesch, Marc Pollefeys, and Roland Siegwart. *Get Out of My Lab: Large-scale, Real-Time Visual-Inertial Localization.* Autonomous Systems Lab and Computer Vision and Geometry Group, Department of Computer Science, ETH Zurich, Switzerland and Google Inc., Mountain View, CA, United States, 2015.
http://people.inf.ethz.ch/sattlert/publications/Lynen2015RSS.pdf [April 29, 2018].

[48] Technische Universität München. Event-based neuromorphic vision.
https://www.nst.ei.tum.de/forschung/neuromorphic-vision/ [April 29, 2018].

[49] Roland Memisevic and Christian Conrad. *Stereopsis via deep learning.* Department of Computer Science, University of Frankfurt, Germany, 2018.
https://www.iro.umontreal.ca/~memisevr/pubs/depthlearn.pdf [April 29, 2018].

[50] eeNewsAutomotive. 3D image sensor uses time-of-flight method.
https://www.zhihu.com/question/37920173/answer/77711032 [April 29, 2018].

[51] Imerso. Capture reality in 3D.
http://imerso.com/ [April 5, 2018].

[52] Matterport. Mobile capture with matterport scenes.
https://matterport.com/matterport-scenes-3/ [April 24, 2018].

[53] Google Play. Constructor developer tool.
http://www.eenewsautomotive.com/content/3d-image-sensor-uses-time-flight-method/
page/0/2 [April 29, 2018].

[54] GSMArena. Asus Zenfone AR ZS571KL.
https://www.gsmarena.com/asus_zenfone_ar_zs571kl-8502.php [March 22, 2018].

[55] Komplett. DJI Phantom 4 PRO.
https://www.komplett.no/product/907550/leker-hobby/
droner-radiostyrt/droner/dji-phantom-4-pro?gclid=
Cj0KCQjwtOLVBRCZARIsADPLtJ0qq5AugdT6PqH9M7zoyyeyjOwZZgokdNZAnGhSoRi5vQTFNtlbEq8aAt
wcB&gclsrc=aw.ds&dclid=CJCWib3-idoCFZeXGAod7BQMpA# [March 22, 2018].

[56] Leica Geosystems. X310.
https://lasers.leica-geosystems.com/x310 [April 3, 2018].

[57] Lenovo. Ideapad 700 (15).
https://www3.lenovo.com/ph/en/laptops/ideapad/ideapad-700-series/
Ideapad-700-15/p/88IP7000671 [April 3, 2018].

[58] BIMobject. Welcome to bimobject® cloud solution.
https://bimobject.com/en [May 6, 2018].

[59] RevitCity. Downloads.
https://www.revitcity.com/downloads.php [May 6, 2018].

# Appendix A

# File formats

- **3ds:** Is a 3D image file format used by Autodesk 3D studio. The file contains material attributes, bitmap references, mesh data, smoothing grouped data, viewport configurations and more. A 3ds file consist of blocks of data called *chunks* that contain an ID and a length description.

- **Bin:** Is a compressed binary file, which is often used as a term for *non-text file*. Such a file contains binary code that is used by various computer applications. Binary code is based on a two-symbol system (often 1 and 0) and is used for applications like computer processor instructions. A 3D model can also be represented as a bin file.

- **Dae:** Is short for *digital asset exchange*. Dae file extension is often used as an identification for Collanda file formats. Collanda, which is short for *COLLAborative Design Activity*. It is an interchange file format for interactive 3D applications. The file format opens for exchanging digital assets among various graphics software applications, by defining an open standard XML schema. This allows exchange of files that is otherwise stored in incompatible formats.

- **Dxf:** Is also an Autodesk software file format. Dxf is short for *Drawing Interchange Format*. The format was made for enabling data interoperability between AutoCAD and other software packages.

- **E57:** Is a vendor-neutral, compact file format for storing point clouds, images and metadata. It is often produced by laser scanners or other 3D imaging systems. ASTM, an international standards organization has specified this as a standard file format and it is documented in the ASTM E2807 standard.

- **Fbx:** Is also known as *Filmbox*. The file format is used to provide interoperability between digital content creation applications. Fbx can be represented as either binary or ASCII data. The file format is owned by Autodesk.

- **Icm:** Is short for *Image colour matching*. The file format contains a colour system pro-

file for applications such as colour printing or scanning.

- **Las:** Is a public file format for interchange of 3D point cloud data. Although it was developed primarily for LIDAR, it supports other point cloud data as well. It is a binary file format specified for point cloud representation.

- **Obj:** Is a simple data-file format that represents 3D geometry alone. This includes the position of each vertex, the UV position of each texture coordinate vertex, vertex normal and faces. The *obj* file has no units but contain information in a human readable comment line. Vertices are stored in a counter-clockwise order by default.

- **Off:** Is short for *Object File Format*. This is a geometry definition file format that contain descriptions of the composing polygons of a 3D object. A typical off file includes: The number of vertices, the number of faces, the number of edges, a list of x, y and z coordinates and a list of included faces and their normal vector.

- **Ply:** Is a file format known as the *Polygon File Format*, or the *Stanford Triangle Format*. The file format was principally designed to store 3-dimensional data from 3D scanners. Properties such as colour, transparency, surface normal, texture coordinates and data confidence values can be stored in the file.

- **Pov:** Is a 3D graphic file format, which is short for *Persistence of Vision*. The file format supports shapes, colours, textures, rotations, light, viewing perspectives and other 3D image properties. It is stored in an ASCII text format.

- **Prj:** Stands for *ProJect*. The file format is a generic extension project file used for many applications. These files are often used to store data, settings and references to other files in the project.

- **Pts:** Is short for *3D Points File*. The format is often used when exporting final registered point clouds. It is often described as a "dumb format", because it does not retain any original scan or registration information.

- **Rcp:** Is a project file format that points to several *rcs* files. These files often contain additional information such as scan regions, recorded distances and annotations.

- **Rcs:** Is a file that contain spatially indexed point cloud data.

- **Rvt:** Is a Revit project file. This file format contains various kinds of data like architectural details, elevations, floor plans and project settings.

- **Shx:** Is better known as a *Shapefile*. The format is a popular geospatial vector data format for Geographic information systems. It can spatially describe vector features like points, lines and polygons.

- **Stl:** Is an abbreviation of *stereolithography*. It is a file format often used by CAD software. The file format is widely used for rapid prototyping, computer-aided manufac-

turing and 3D printing. A *stl* file uses a 3D cartesian coordinate system to describe raw, unstructured triangulated surfaces by a unit normal and vertices.

- **U3d:** Is short for *Universal 3D* and is a compressed file format used for various 3D computer graphics data. The format was standardized by Ecma International in 2005 as ECMA-363.

- **Vrml** 2.0: Is short for *Viritual Reality Modelling Language, version 2.0.* The file format is standardized for representing 3D interactive vector graphics, designed particularly for the world wide web. A vrlm uses a text file format, were vertices and edges of a 3D polygon can be specified along with the surface colour, UV-mapped textures, shininess, transparency and more.

- **Vtk:** Is short for *Visualization Tookit.* This is a toolkit for reading and writing popular data file formats. There are also some specific *vtk* file formats created as a simple method to communicate between various software's that do not support regular visualization file formats.

- **X3d:** is an ISO standard for declaratively representing 3D computer graphics. This include several file formats like *xml, vrml* and *bin. X3d* file formats are used for several applications, mostly in open-sourced software.

- **Xyz:** Is a file format for storing coordinates. This file format is most famous for saving molecular coordinates, whereas the xyz file contain the coordinates atoms in a system. It can also be used for storing point clouds, where each point is stored with x, y and z-coordinates.