

UNIVERSITY OF TARTU

Faculty of Science and Technology

Institute of Technology

Kwasi Akuamoah Boateng

# **Digital Twin of a Teaching and Learning Robotics Lab**

Bachelor's Thesis (12 ECTS)

Curriculum Science and Technology

Supervisor:

Associate Professor Karl Kruusamäe

Tartu 2021

## **Abstract**

The advancing technologies of Industry 4.0, which includes digital twins, is gaining ground and becoming more popular in many industrial sectors. In the manufacturing industry, digital twins are used, ranging from simulation to product optimisation. This work focuses on using LiDAR data, SLAM algorithms and basic measure tape for developing a digital twin environment in the open-source platform Gazebo backed by ROS, which scientists, engineers, and students will use to streamline development process, for educational purposes and many more. The work results show a digital replica of specific areas of the Institute of Technology, where multiple robots can be integrated and controlled. Such a platform creates a foundation for improving distance learning and safe initial system testing.

**CERCS:** T120 Systems engineering, computer technology, T125 Automation, robotics, control engineering

**Keywords:** Industry 4.0; digital twins; LiDAR; SLAM algorithms; Gazebo; ROS; streamline processes; multiple robots; system testing

## **Lühikokkuvõte**

### **Robotite õpetamise ja õppimise labori digitaalne kaksik**

Paljudes tööstussektorites on populaarsust kogumas Tööstuse 4.0 arenevad tehnoloogiad, mille alla kuuluvad ka digitaalsed kaksikud. Tööstuses kasutatakse digitaalseid kaksikuid simulatsioonist kuni toote optimeerimiseni. See töö keskendub LiDAR andmete, SLAM-algoritmide ja mõõdulindi kasutamisele digitaalse kaksikeskkonna arendamiseks avatud lähtekoodiga platvormil Gazebo, mida toetab ROS (Robot Operating System). Teadlased, insenerid ja üliõpilased kasutavad seda arendusprotsessi sujuvamaks muutmiseks nii hariduslikel kui ka muudel eesmärkidel. Töö tulemused näitavad tehnoloogiainstituudi konkreetsete valdkondade digitaalset koopiati, mille kaudu saab integreerida ja juhtida mitut robotit. Selline platvorm loob aluse kaugõppe parandamiseks ja süsteemi ohutuks testimiseks.

**CERCS:** T120 Süsteemitehnoloogia, arvutitehnoloogia, T125 Automatiseerimine, robotika, juhtimistehnika

**Märksõnad:** Tööstus 4.0, digitaalne kaksik, LiDAR, SLAM, Gazebo, Roboti operatsioonisüsteem (ROS), sujuvam protsess, mitmed robotid, süsteemi testimine.

# Table of contents

<b>Abstract</b>	<b>2</b>
<b>Lühikokkuvõte</b>	<b>2</b>
<b>Table of contents</b>	<b>4</b>
<b>1. Introduction</b>	<b>6</b>
1.1. Background	6
1.2. Problem Statement, Motivation, and Objective	6
<b>2. Literature Review</b>	<b>8</b>
2.1. Digital Twin Technology	8
2.2. Applications of DT	10
2.2.1. Manufacturing	10
2.2.2. Healthcare	11
2.3. Tools	12
2.3.1. Gazebo Simulator	12
2.3.2. CoppeliaSim	13
2.3.3. Unity	13
<b>3. Purpose and Requirements</b>	<b>14</b>
3.1. Objective	14
3.2. Functional requirements	14
3.3. System requirements	14
<b>4. Technical Solution</b>	<b>15</b>
4.1. Metre rule/measuring tape method	16
4.1.1. Taking measurements	16
4.1.2. Developing a Gazebo model from CAD model	16
4.1.3 Results	16
4.2. 3D PointCloud	17
4.2.1. Data Gathering	17
4.2.2. Rosbag to point cloud, creating a mesh	18

4.2.3. Results	18
4.3. 2D SLAM	19
4.3.1. Mapping	20
4.3.2. Developing a mesh from the map	21
4.3.3 Results	21
<b>5. Analysis of Results</b>	<b>23</b>
<b>6. Discussion and Conclusions</b>	<b>27</b>
6.1. Known issues	27
6.2. Future Improvements	27
<b>References</b>	<b>29</b>
<b>Extras</b>	<b>33</b>
Annex 1	33
Annex 2	33
Annex 3	33
Annex 4	33
<b>Non-exclusive Licence to Reproduce Thesis and Make Thesis Public</b>	<b>34</b>

# **1. Introduction**

## **1.1. Background**

Traditional simulations have been used successfully for years to solve optimisation problems within production and logistic systems. Simulation helps with understanding what might happen if changes are made to these complex systems. Due to its limitations, including not being precise, a more advanced technology called a digital twin can potentially do much more.

Digital Twin (DT) is a computer-generated equivalent of a physical system. DTs provide engineers with virtual tools that allow them to explore and assess the physical object's physical assets, processes, and systems. With this technology, it is possible to get an accurate view of present instances and what will happen in the future [1].

The use of DTs in the modern world is growing every day [2]–[4]. Over the last years, virtual models of real objects were used for remote monitoring, prototyping, product design, and simulation [2], [5]–[7].

Today, we are engaged in the fourth industrial revolution known as "Industry 4.0," which is the gateway to cyber-physical systems (CPS) and intelligent automation, like robotics. Industry 4.0, together with DTs is transitioning traditional analog processes to digital ones, hence rapidly increasing digital manufacturing [8].

Furthermore, DTs have become a popular research topic in computer science and technology [9]. However, the advent of the Internet of Things (IoT) which serves as the medium through which data is communicated from the digital device to its physical counterpart in either direction has given a newer insight into the concept in recent years [10]–[12].

## **1.2. Problem Statement, Motivation, and Objective**

The Intelligent Materials and Systems (IMS) Laboratory is an interdisciplinary research group at the University of Tartu's Institute of Technology that was founded in 2003. IMS Robotics is a division of IMS Lab that specialises in research, development and education of robotics. Despite

the availability of a diverse selection of physical robots at the IMS Robotics Lab, there is a need for a fully integrated accurate digital twin of the premises with all the robots. Such a digital twin is needed for, e.g.,

- 1) streamlining development process - any developed algorithms can be validated first in simulation and multiple people can work on the same setup simultaneously,
- 2) scaling up the learning experience - more novice students can have access to learning robotics with the help of a simulated environment,
- 3) safe initial system testing - automated unit tests of any developed functionality can be run extensively on the simulated scenarios.

This thesis aims to develop a digital twin of the robotics labs at the Institute of Technology, University of Tartu, to meet the needs of students, researchers, and engineers.

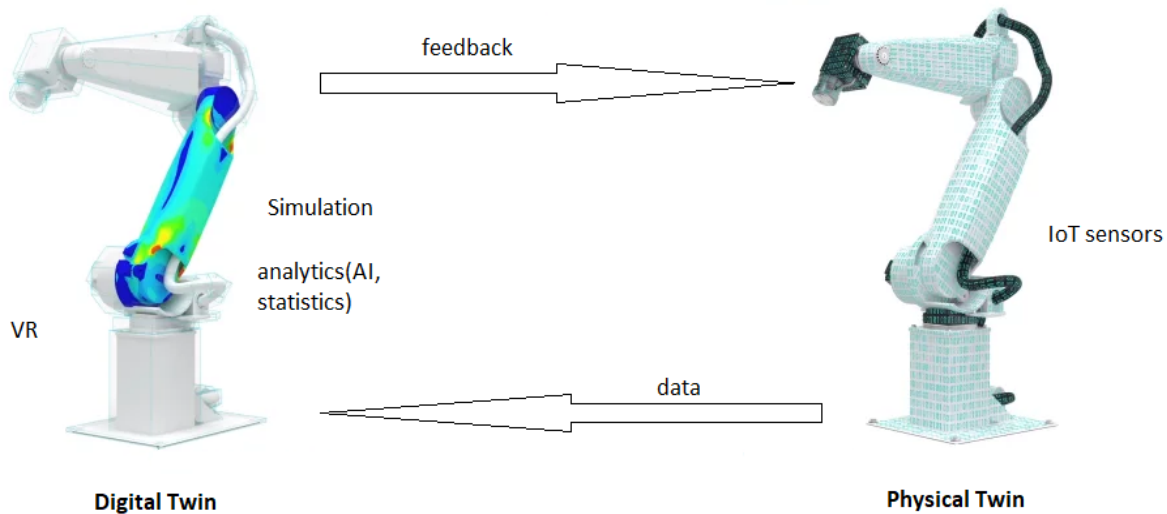
## 2. Literature Review

This chapter discusses digital twin technology (section 2.1), its applications (section 2.2), and the software tools (section 2.3) used to develop a digital twin environment in robotics.

### 2.1. Digital Twin Technology

The increasing availability and ubiquity of real-time operational data, as well as the advancement of artificial intelligence (AI) implementation capabilities in learning and reasoning, are all contributing to the realisation of a vision of physical products or processes with virtual representations that evolve throughout their entire life cycle. Such so-called virtual counterparts are called Digital Twins (DT).

A DT is a virtual representation of a physical device, system, or process [11], [13], [14]. As shown in Figure 1, DTs are composed of models, functionality, and communication interfaces with their physical counterparts [15]. A DT follows its physical twin's lifecycle to monitor, control, and optimise its operations. It continuously predicts future states (for example, defects, damages, failures) and allows new configurations to be simulated and tested to prevent maintenance activities from taking place [16].



**Figure 1.** Digital twin concept [17]

With the use of IoT (Internet of Things) devices [18], [19], and web services [20], data are continuously exchanged and updated in real-time. The physical object or process updates its environment with real-time data and is constantly aware of its physical surroundings [16]. In order to function fully, the DT has specific network requirements to be satisfied [1]. With large amounts of big data being collected from hundreds of possible sensors simultaneously, high bandwidth and high data rate are required to transmit them all at once. As such, 4G and, above all, 5G networks must be considered [1].

AI and machine learning play a massive role in DT technology in recent advancements [21]–[23]. Advanced AI algorithms can predict and prescribe actions in the physical world in real-time to reconfigure and reduce issues with production downtime without humans' assistance [24].

DT concept should not be limited to the real-time exchange of data between the physical device and its virtual counterpart [13]. Alternatively, the concept involves having a digital model of an actual asset that either connects to its physical system to obtain data automatically or depend on manual data input (for example, The Living Heart) [13], [25]. The virtual replica uses the data from the physical system to develop and derive solutions relevant to the real system [5], [6].

To create a fully functional DT, Tao et al. [26] presented a six-step requirement procedure in their work which includes:

1. Create a virtual representation of a physical product using CAD or 3D modelling.
2. Processing data to aid design decision-making.
3. Simulate physical systems in virtual environments.
4. Test and calibrate the physical product to work in tandem with its DT.
5. Establish a communication interface between the physical and virtual system.
6. Collect relevant data for continuous system integration.

Since the technology is in its growing stages, Chen et al. [27] describe DTs for automobiles and traffic management. The paper investigates the difficulties of driving, demonstrating the need for more data flow within the vehicle used as well as a connection to other vehicles in the vicinity.

Chen et al. [27] describe a framework that employs a DT in conjunction with learning algorithms that monitor and analyse feedback based on user behaviour. The algorithms enable a driver's real-time digital behavioural twin, providing warnings and instructions on driving more safely to reduce risk [27].

## **2.2. Applications of DT**

### **2.2.1. Manufacturing**

In the field of manufacturing, DT technology is actively progressing. The current developments follow the Industry 4.0 concept, which utilises the connectivity of devices to make the idea of a DT a reality for manufacturing processes [2], [11], [20]. Primary use cases of DTs in manufacturing are quality management, product design, system planning and plant layout, and product development.

Companies may use DTs to detect equipment defects before they halt manufacturing, enabling repairs to be completed sooner and at a lower cost. DTs have the potential to increase connectivity and feedback between devices, improving reliability and performance.

In China, a Siemens factory that produces electronic components uses the DT technology for optimising production [28]. The factory records and monitors all production lines. The primary goal of the DT is to process and analyse vast volumes of production process data to ensure production efficiency and product quality [28].

The use of DTs for Bentley Systems allows the company to explore vast manufacturing plants and offshore facilities virtually as if they were simply travelling around the facilities [29]. Currently, TESLA intends to demonstrate DT technology's capabilities by creating a DT for each constructed car, allowing synchronous data transfer between the vehicle and the plant [4].

In automated assembly lines, part mating is a critical phase in the manufacturing process. Many facets of mating two pieces can be boiled down to a peg-in-hole situation. Jamming may occur as parts are fitted to each other, resulting in assembly failure or damage to the components, the robot, or the environment. As a result, for creating and testing new algorithms, an inherently

secure environment is needed. Grinshpun et al. [30] suggested a technique for designing algorithms for robot control. They developed and validated a peg-in-hole insertion algorithm. The DT was used to verify and validate this algorithm inside the simulation before transferring it to the physical robot.

### **2.2.2. Healthcare**

In the healthcare sector, DTs have been first used for product or equipment prognostics to improve the accuracy and efficiency of many medical devices [31]. By optimising patient treatment with remote monitoring and advanced simulation of the human body, DTs revolutionise healthcare procedures and hospital administration. A few healthcare startups have progressively applied the technology. An example is Babylon Health [32], an AI-powered app that creates a DT based on a questionnaire about family history and lifestyle. The DT is capable of predicting current health status as well as risk factors for future conditions while providing practical ways to stay healthy [32].

Alternatively, Sim&Cure has created a patient-based digital twin for treating brain aneurysms in collaboration with ANSYS [16], [33], [34]. Endovascular reconstruction, in which a catheter-guided implant is used to stabilise weakened arteries and relieve strain on aneurysms caused by irregular blood flow, is progressing thanks to the twin [16], [33], [34]. A surgeon can better consider the expected effects of implant installation by generating patient-specific 10-20 second simulations [33]. Although 10% of endovascular therapies typically require follow-up procedures, no additional intervention was needed in any of the digital twin procedures at the time of the study [33].

Furthermore, The Living Heart, developed by the French software firm Dassault Systèmes was launched in May 2015 and is readily available for testing [25]. The modelled heart was the first DT of organs to consider all facets of the organ's operation (including mechanics, blood flow and electrical pulses) [16]. The program requires a 2D scan (CT or MRI) input, which is converted into a trustworthy 3D organ model [16], [25]. The heart model enables doctors to run hypothetical simulations (e.g., inserting a pacemaker or reversing the heart's chamber) to forecast patients' results or make decisions [25].

On the other hand, DT is about more than mere data management and analysis. It is a game-changing approach to building a dynamic computer infrastructure that connects various aspects of a whole system, whether it is an actual patient and their body's structures or the healthcare ecosystem's treatment systems. The aim is to extract actionable lessons that will allow for better decision-making and, as a result, better patient outcomes [31]–[34].

## **2.3. Tools**

The following tools are simulation-based software. These applications are known for their powerful simulation capabilities. Alternatively, these software tools can be coupled with IoT devices and other software to establish a communication interface between the physical and DT systems.

### **2.3.1. Gazebo Simulator**

Gazebo is a powerful open-source physics-based 3D simulator capable of accurately and efficiently simulating populations of robots in complex indoor and outdoor environments [35]–[37].

Gazebo is the default simulator used in the ROS (Robot Operating System) framework. Gazebo has plugins that provide easy ROS communication methods. One significant advantage of using Gazebo over other simulators is its large base of community-developed plugins and code [38].

This simulator has four different physics engines: ODE, Bullet, Simbody, and DART [35]. It would be sufficient to invoke the preferred engine during the launch in the scripts that activate the packages to be executed with their respective parameters. Gazebo performs well in simulating multiple instances of robots in one session [38]. Despite its superior processing power, real-time editing of models is impossible in Gazebo; external 3D-modelling tools like Solidworks or blender are required [38].

In [39], Niki et al. set up a DT environment using the Gazebo simulator to explore digital world modelling strategies in hybrid production processes for allowing system reconfiguration through a shared environment and process perception. The team loaded virtual models of the physical

robot in the simulation. Communication was made possible by the ROS framework. The DT environment updated and adapted to instances in real-time (i.e. a new object added to the physical world).

### **2.3.2. CoppeliaSim**

CoppeliaSim, previously known as Virtual Robot Experimentation Platform (V-REP), is one of the most potent robot simulation tools available, with a plethora of features that are extremely useful for robotics enthusiasts. In contrast to Gazebo, CoppeliaSim is free for educational purposes only but has limitations for commercial use [38]. CoppeliaSim has the advantage of its easy to use features. Models can be easily modified within the simulator, which is very convenient and useful. Deep knowledge of XML is not required. Like Gazebo, CoppeliaSim has several physics engines such as ODE, Bullet, Vortex and Newton Dynamics [40].

Boris' team [41] used V-REP to find inspection paths automatically. The team matches the movements of a virtual Kuka KR16 with a real one.

### **2.3.3. Unity**

Unity (commonly known as Unity3D) is a game engine that was developed by Unity Technologies in 2005. Since then, the app has grown in popularity as a game engine. Despite its background as a game engine, it has also been used as a robotics simulator [42]. Unity uses the PhysX physics engine, which offers the possibility to define joints in both Cartesian and joint coordinates [43]. Besides, Unity allows the development of immersive virtual reality (VR) [44].

Pérez et al. [45] utilised the outstanding capabilities of Unity3D to create a VR representation of a DT. In their work, the to-be virtual scenario is first scanned with a 3D scanner, processed and polished using the Blender software, and converted into a VR environment.

## **3. Purpose and Requirements**

### **3.1. Objective**

The aim of this bachelor's thesis is:

- To develop a digital replica environment of the robotics labs at the Institute of Technology in Gazebo.
- To integrate and simulate robots available at the Institute of Technology in the Gazebo world.
- To test existing Gazebo packages of off-the-shelf robots.

### **3.2. Functional requirements**

1. Realistic representation of room 329 at the Institute of Technology
  - 1.1. Include robots UR5, Clearpath Jackal, Robotont, TurtleBot3
2. Realistic representation of the 3rd-floor hallway of the Institute of Technology
  - 2.1. The hallway is constructed from the sensor data.

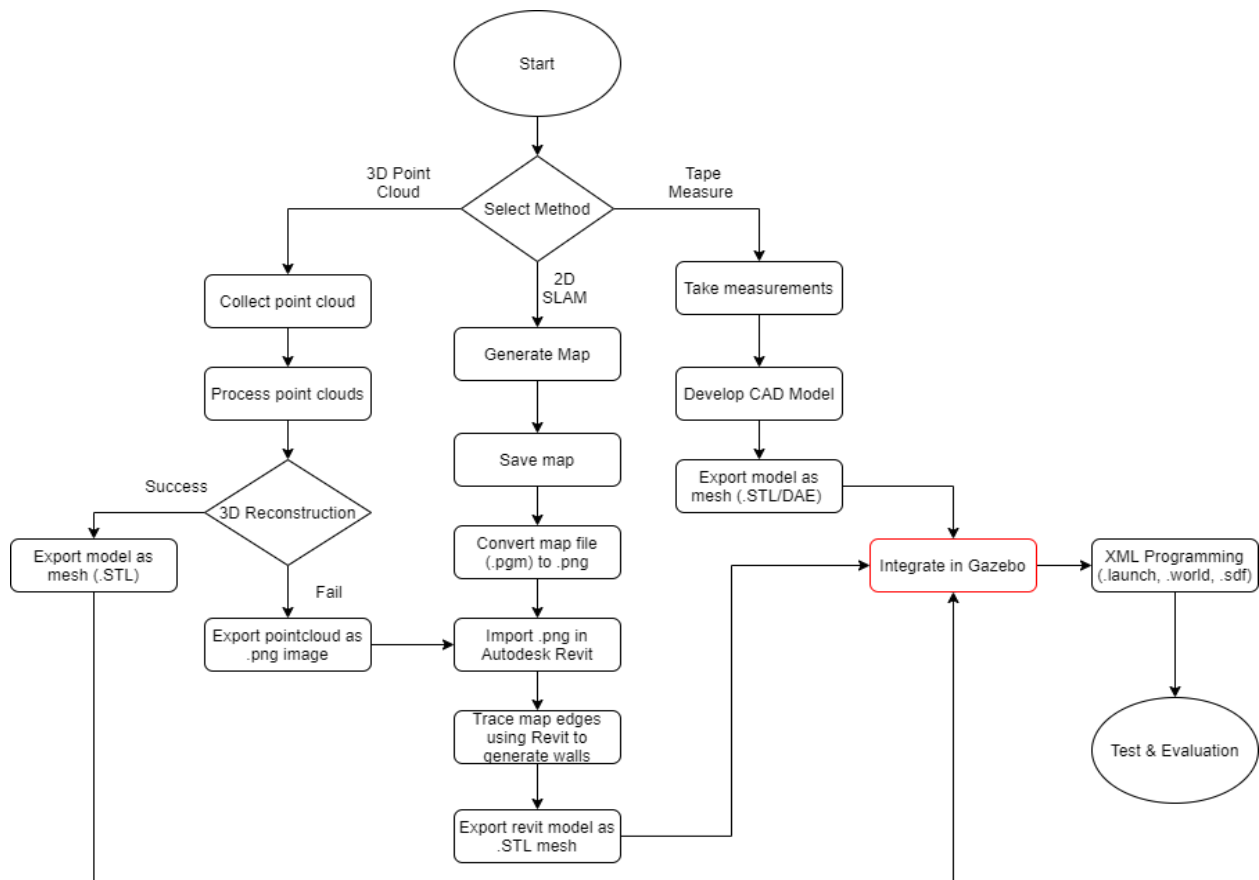
### **3.3. System requirements**

- Gazebo ver 9.x or higher
- Compatibility with ROS Melodic

## 4. Technical Solution

The following section describes methods used in developing 3D models to use in the Gazebo simulator. Figure 2 depicts the entire process in this work. Three data collection methods were involved during the process. Chapter 4.1 presents human efforts of taking measurements, while chapter 4.2 and 4.3 describe robotic sensor technology methods.

Models can be dynamically loaded into simulation either programmatically or via the Gazebo GUI. These models must abide by a specific directory and file structure. A model directory also includes the model's SDF (Simulation Description Format) as well as any materials, meshes (STL or DAE format), and plugins.



**Figure 2.** Flowchart of the 3 methods used in this thesis to generate digital twins

## **4.1. Metre rule/measuring tape method**

### **4.1.1. Taking measurements**

In order to make a digital replica of a real-life object, it is necessary to identify the location and orientation of the object. The size and the weight of the object must be taken into consideration.

In this work, measuring instruments such as a metre rule and tape measure were used interchangeably to measure the length, width and height of Room 329. Measurements of furniture were also taken. By utilising these measurement data, a digital copy of the real-world environment was developed (Figure 3).

### **4.1.2. Developing a Gazebo model from CAD model**

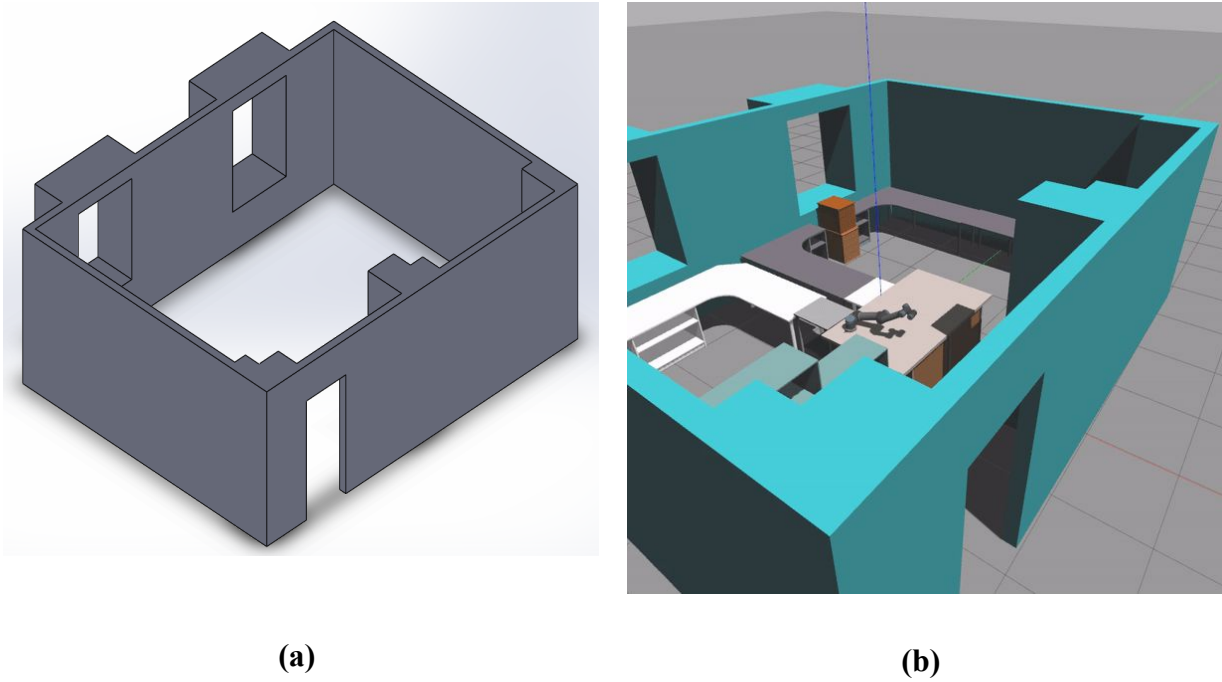
Many CAD software in the market can create a digital model using the data collected. In Solidworks, the *SolidWorks to URDF exporter* addon can convert 3D CAD models into a URDF (Unified Robot Description Format), a format compatible with Gazebo. The addon exports a folder containing a mesh file in STL format and its URDF.

URDF formats must be converted to SDF formats to represent objects (furniture etc.) as Gazebo models. Typically, URDF is used to describe the kinematics of a robot.

For the SDF formats to be spawned in the Gazebo GUI, the folders containing the SDF models must reside in `./gazebo/models/` found in the 'Home' folder in Ubuntu by default. However, as a requirement, these model files were added to the ROS package of Annex 1. To implement this strategy, the *package.xml* file in the `unitartu_tech_gazebo` package was edited to depend on `gazebo_ros` (Annex 4). The package `gazebo_ros` offers ROS plugins for message and service publishers to communicate with Gazebo through ROS.

### **4.1.3 Results**

After processing and transferring the measured data to build a 3D CAD model of room 329, the result can be seen in Figure 3a. The resulting DT of lab room 329 can be seen in Figure 3b.



**Figure 3.** CAD model of the robotics room(a), Digital twin of room 329 in Gazebo (b)

## 4.2. 3D PointCloud

A point cloud is an unstructured collection of points representing an object. It contains information about an object's location in 3D space, normal vector, colour, transparency and size [46]. To obtain a point cloud of a physical object, the use of a sensor, specifically a laser scanner or lidar, is required.

In this experimental setup for this experiment is the Jackal UGV. The Ouster OS1 64-beam lidar and other external sensors, such as a camera or an Inertial Measurement Unit (IMU), are mounted on the robot. The lidar is the critical range sensor in this work for mapping and localisation. This work does not make use of the camera data.

### 4.2.1. Data Gathering

ROS packages that were used in this experiment are given in Annex 2. The robot Jackal was guided manually with an android mobile app (ROS-Mobile [47]) in the hallway. The entire mapping process is recorded into a rosbag [48] (.bag) file for later processing.

#### **4.2.2. Rosbag to point cloud, creating a mesh**

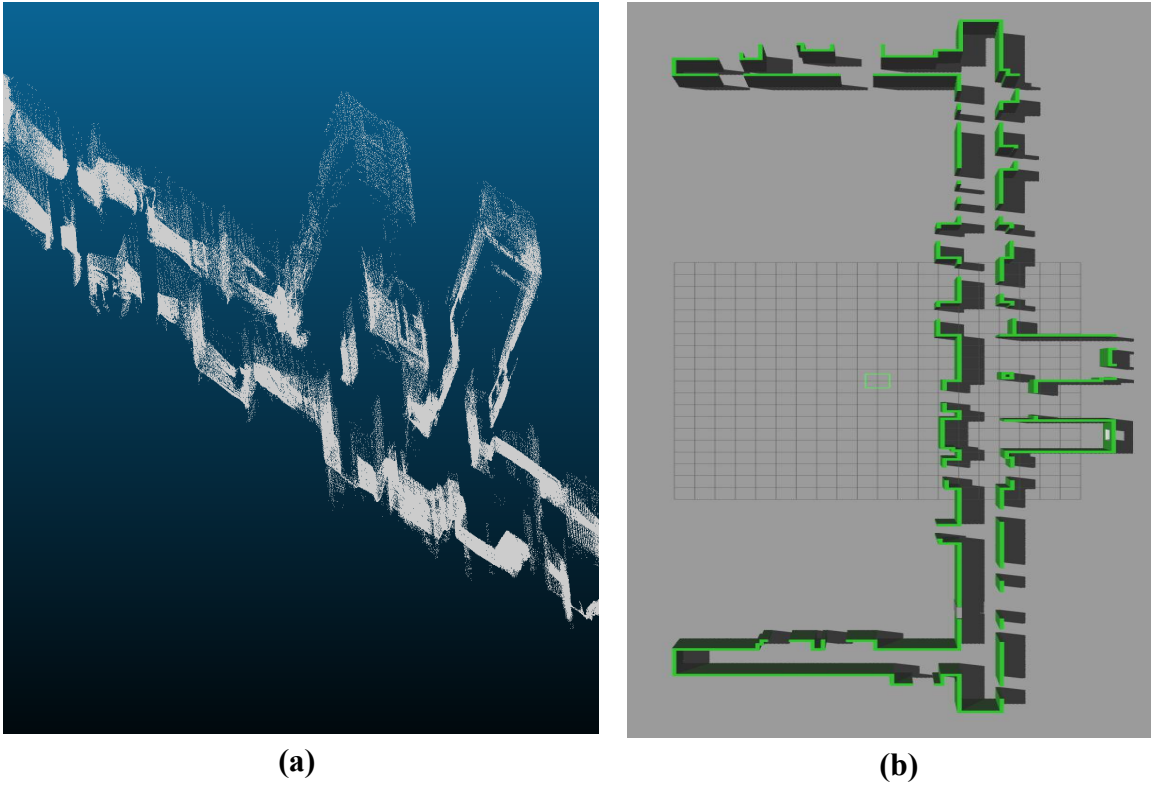
Point cloud messages saved to a specific topic in the *.bag* file are extracted by using the `bag_to_pcd` node in `pcl_ros` [provide wiki.ros REF] package. The exact command is:

```
roslaunch pcl_ros bag_to_pcd <input_file.bag> <topic> <output_directory>
```

The PCD (Point Cloud Data) files obtained are imported into CloudCompare [49]- a software for processing 3D point clouds and meshes. In CloudCompare, the point cloud data is processed by implementing options such as segmentation, registering or aligning point clouds (ICP) and noise reduction. A suitable mesh can be created; however, this work does not include mesh creation with point cloud data due to limited knowledge. A proposed method can be seen in Figure 2. A link in Annex 3 describes the detailed procedure to import the created mesh into Gazebo.

#### **4.2.3. Results**

The findings obtained in this work were achieved using the map to mesh approach. More specifically, the Autodesk Revit approach is described in Figure 2. In Figure 4, PCD files from the *.bag* file were extracted to generate a mesh and imported into Gazebo.



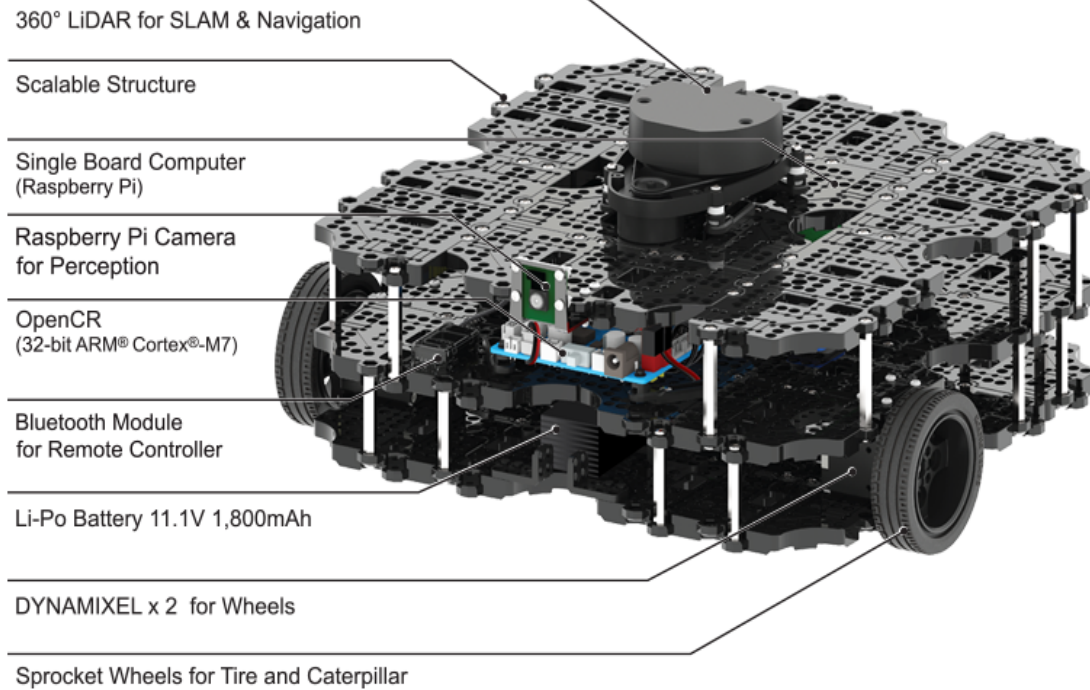
**Figure 4.** PCD visualization in CloudCompare (a), Gazebo model of the hallway (b)

### 4.3. 2D SLAM

SLAM (Simultaneous Localization & Mapping) is a technique used to build a map of an environment while at the same time keeping track of the current location. The laser scanner mounted on the robot (Figure 5) makes it possible for the robot to sense its environment and collect a set of data around it to use for SLAM.

In ROS, there are some options available for performing 2D SLAM. Figure 5 depicts the robot that was used in implementing 2D SLAM.

## Waffle Pi



**Figure 5.** The experimental robot (Turtlebot3) [50]

### 4.3.1. Mapping

The mapping operation can be performed in two ways, either online or offline. The offline method depends on already recorded data. However, in the online method, it is possible to record the data for later use. To record the data, one can use the *rosviz* tool. One can later play the recorded data using the same tool and perform mapping with the published data.

The TurtleBot3 robot (Figure 5) offers SLAM methods such as *gmapping*, *hector*, *cartographer* and many more [50]. In the experiment, two methods were used; *cartographer* and *gmapping*. To perform SLAM using either the *gmapping* or *cartographer* algorithm, the chosen method can be set from a command-line script.

The entire mapping method can be visualised using Rviz. Once the environment map is created in Rviz, it is important to save the resulting map. The *map\_server* [51] offers a command-line

tool for saving the map into the computer. The generated map comprises two files, the *.pgm* image file containing the occupancy grid (e.g. Figure 6) and a *.yaml* file containing the map's properties.

### **4.3.2. Developing a mesh from the map**

The first step towards creating a 3D model from a generated map is to convert the *.pgm* image file to a suitable extension such as *.png* by using the GIMP software or other image editing tools. The next step is to use Autodesk's Revit [52] application. In Revit, walls can be traced by importing the PNG image. The Revit model of the hallway is exported to STL format using the *STL Export* add-on.

The final step is to develop a Gazebo model by using the STL mesh obtained. The detailed procedure can be followed in the link given in Annex 3.

### **4.3.3 Results**

Figure 6 shows experimental results of *cartographer* and *gmapping* methods for mapping. These results were collected on the 3rd-floor hallway of the Institute of Technology. The first map (Figure 6a) was built using the *cartographer* slam method. However, the other two maps (Figure 6b-c) were built using the *gmapping* algorithm.

The default presets were used in both slam methods. The *gmapping* results are inappropriate because they do not reflect the true layout of the corridor. Figure 7 shows results of hallway mesh imported in Gazebo. The hallway mesh was developed from the *cartographer* map (Figure 6a).

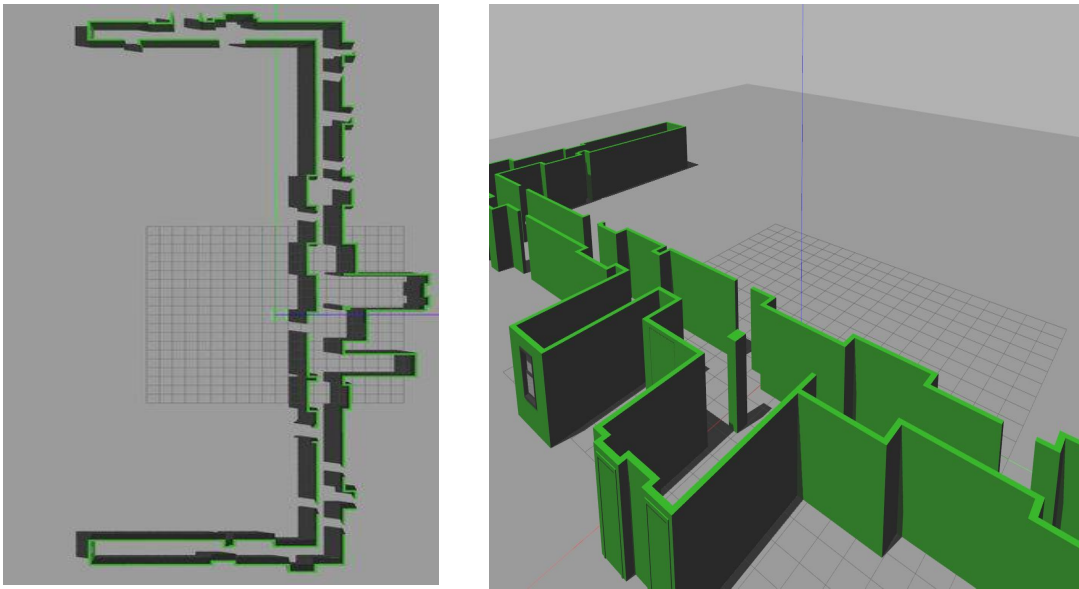


(a)

(b)

(c)

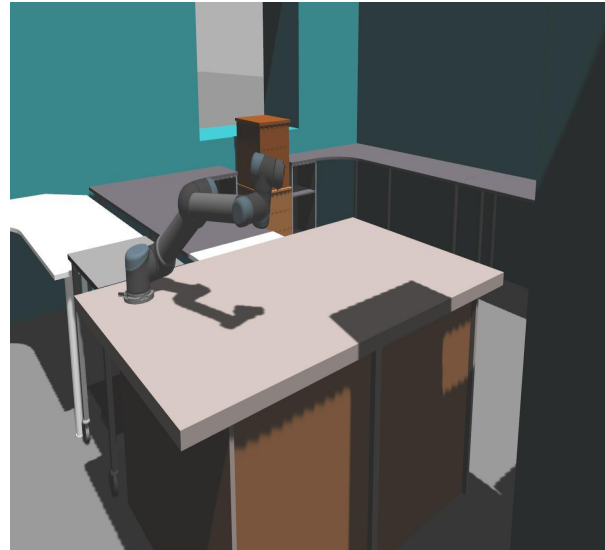
**Figure 6.** Maps built using cartographer (a) and gmapping (b), (c) SLAM methods



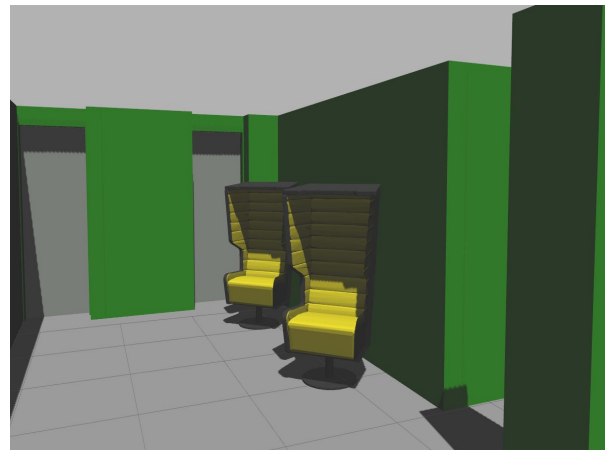
**Figure 7.** Gazebo model obtained using the map from cartographer SLAM

## 5. Analysis of Results

At the end of the experiment, the side-by-side comparisons of physical and digitally-developed environments can be seen in Figure 8. The final digital twin with different robots is depicted in Figure 10.



(a)



(b)

**Figure 8.** Real-life vs Digital Twin. (a) Lab room 329, (b) Hallway

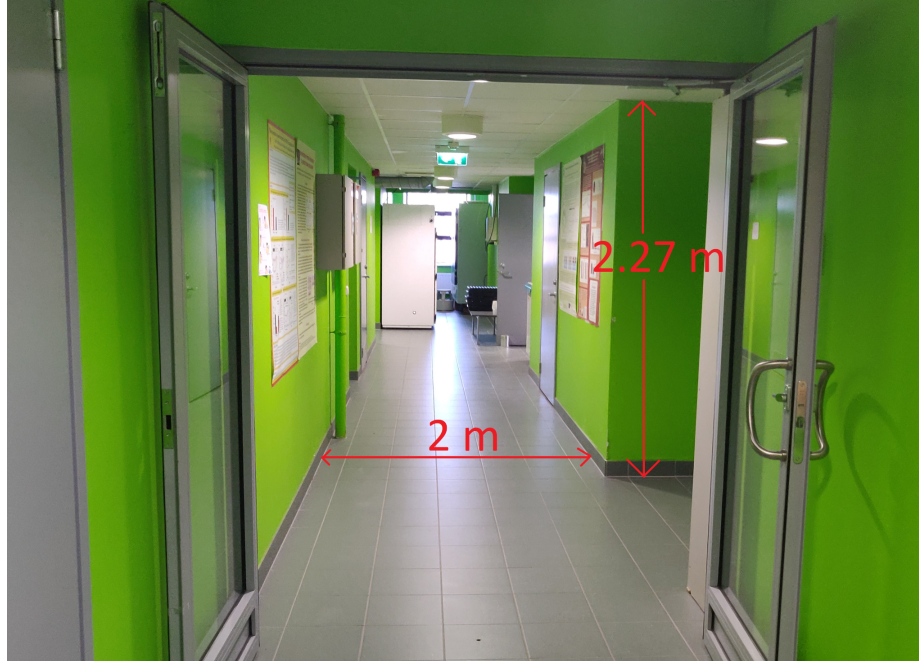
The decision to select the best resulting mesh is dependent on the average accuracy of measurement methods (Table 1). The results were obtained by comparing side-by-side measurements of physical objects with their digital counterparts. Hence, calculating the average

difference in measurements (i.e. height, width and length) of the physical environment and their 3D mesh models.

**Table 1.** Measurements and method accuracy

Reference model measurements / m		Method	Hallway mesh measurements / m				Measurement accuracy / $\pm$ m	
Height	Corridor width							
2.27	2		$L_{\text{corner}}$	$L_{\text{centre}}$	Corridor Width	Height	Corridor Width	Height
		3D Point Cloud	15.35	50.65	1.8	2.30	0.2	0.03
		2D SLAM	18.61	59.76	2	2.30	0	0.03

With the metre rule measurements, the resulting inaccuracy is due to difficulties in reaching higher heights, corners and inaccessible areas. As such, approximate readings were considered. The furniture measurements and that of the room have an interval of  $\pm 0.004$  m and  $\pm 0.025$  m, respectively. These differences are minimal to significantly reflect when comparing the physical environment to its digital twin. Based on that,  $\sim 30$  mm offset is used as a reference model for actual measurements.



**Figure 9.** Actual hallway measurements used as reference

In the SLAM method, the resulting map was used to determine the measurement accuracy of the hallway. The lengths of the centre hallway (represented as  $L_{\text{centre}}$ ) and one corner to the intersection of the centre hallway (represented as  $L_{\text{corner}}$ ) were estimated on the cartographer map with the GIMP app and recorded as 59.76 m and 18.61 m, respectively. The corridor width measured  $\sim 1.992 - 2.087$  m. However, 2.0 m width was adopted in generating a 3D mesh model.

While the point cloud method was difficult to work with, the  $L_{\text{corner}}$  and  $L_{\text{centre}}$  measured 15.35 m and 50.65 m, respectively. A reference used for calculating the inaccuracies can be found in Figure 9.

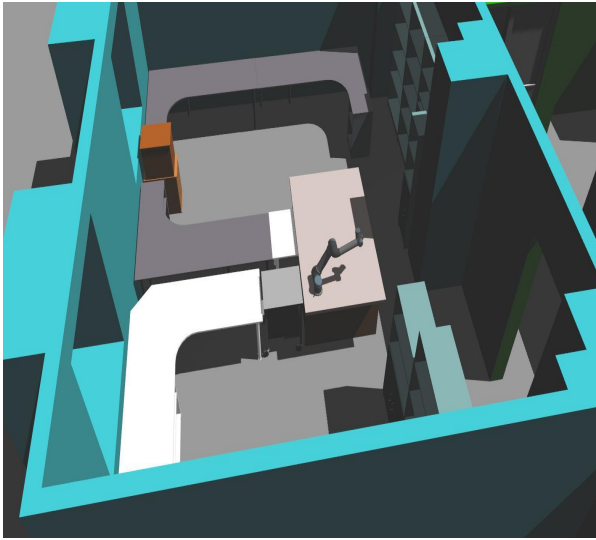
The formula for calculating the measurement accuracy is given below;

$$M_{\text{accuracy}} = \pm |R_{\text{object}} - M_{\text{object}}|$$

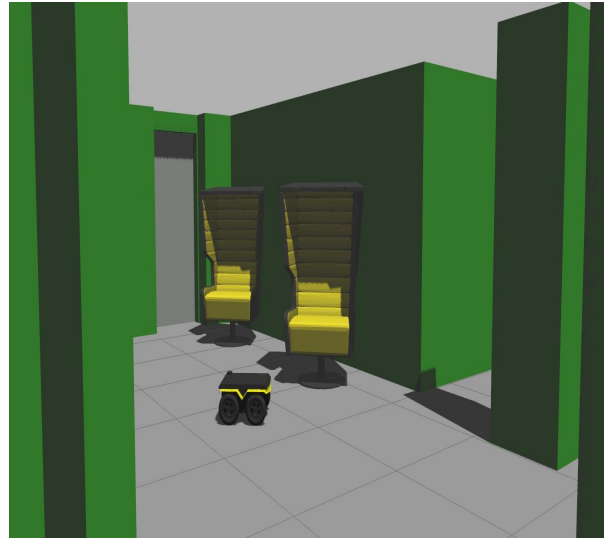
Where,  $M_{\text{accuracy}}$  = measurement accuracy

$R_{\text{object}}$  = measurement of the real environment

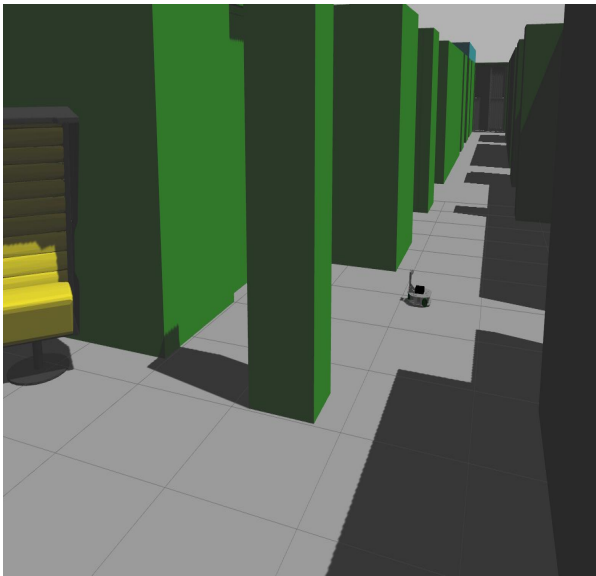
$M_{\text{object}}$  = measurement of the mesh object



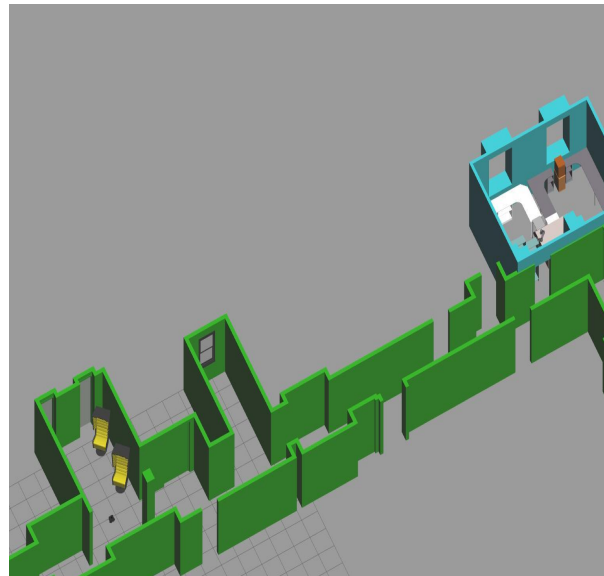
(a)



(b)



(c)



(d)

**Figure 10.** Digital Twin of the hallway and lab room 329 displayed in Gazebo simulator. (a) Room 329 with a UR5 robot mounted on the table, (b) Jackal robot at the entrance of the hallway, (c) Robotont and (d) orthographic view of the digital twin with TurtleBot3 and UR5 robots.

## **6. Discussion and Conclusions**

With the work done on this thesis, all the necessary steps involved in developing a DT model has been covered. More specifically, a thorough review of the literature was undertaken, laying the groundwork for future research and development.

The hallway model from the 3D point cloud method was not used for the final digital twin because it produced highly inaccurate measurements (Table 1).

As of now, the digital twin can be accessed from the link given in Annex 1. Instructions for configuring the digital twin can also be found in the GitHub repository. While code development is ongoing, the current digital twin can integrate and control two specific robots simultaneously (Figure 9d). Additionally, separate launch files are available for the following robots; UR5, Clearpath Jackal, Robotont and TurtleBot3.

In conclusion, digital twins of room 329 and the hallway at the Institute of Technology were developed using data from sensors and measuring instruments. When used properly, DT technology will increase enthusiasm for studying and enhance learning.

### **6.1. Known issues**

During the course of developing the DT, the following issues gradually occur:

- Mounting and spawning the UR5 robot with the default presets causes a collision. This is due to the wrist that touches the lab table.
- There are several problems with successfully building the catkin workspace on the first try.
- Multi-robot spawning for Robotont and Jackal is problematic.

### **6.2. Future Improvements**

This project can be extended to include a communication interface between the real robots and their DTs in the Gazebo world.

The next step in development is to create a VR interface where students, researchers and engineers could interact intuitively and control the DT robots in an immersive way.

The project code development is being continued to improve stability and add new useful features. Also, photogrammetry and 3D scanning methods could automate the generation of a mesh for fast integration in the digital world.

## References

- [1] A. R. Al-Ali, R. Gupta, T. Zaman Batool, T. Landolsi, F. Aloul, and A. Al Nabulsi, “Digital Twin Conceptual Model within the Context of Internet of Things,” *Future Internet*, vol. 12, no. 10, Art. no. 10, Oct. 2020, doi: 10.3390/fi12100163.
- [2] “Digital and Visual Twins: FANUC’s Vision for the Future of Manufacturing | FANUC America.”  
<https://www.fanucamerica.com/news-resources/articles/digital-and-visual-twins-fanuc-s-vision-for-the-future-of-manufacturing> (accessed Apr. 14, 2021).
- [3] “Robotics Online,” *Robotics Online*.  
[https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Videos/Digital-Twin-Powered-by-FASTSUITE-FANUC-MAKINO/content\\_id/7364](https://www.robotics.org/content-detail.cfm/Industrial-Robotics-Videos/Digital-Twin-Powered-by-FASTSUITE-FANUC-MAKINO/content_id/7364) (accessed Apr. 14, 2021).
- [4] Thinkwik, “Digital Twin, A Technology That Changed The World,” *Medium*, Nov. 28, 2018.  
<https://medium.com/@thinkwik/digital-twin-a-technology-that-changed-the-world-5ff5fa685b25> (accessed Apr. 13, 2021).
- [5] C. Koulamas and A. Kalogeras, “Cyber-Physical Systems and Digital Twins in the Industrial Internet of Things [Cyber-Physical Systems],” *Computer*, vol. 51, no. 11, pp. 95–98, Nov. 2018, doi: 10.1109/MC.2018.2876181.
- [6] T. Gabor, L. Belzner, M. Kiermeier, M. T. Beck, and A. Neitz, “A Simulation-Based Architecture for Smart Cyber-Physical Systems,” in *2016 IEEE International Conference on Autonomic Computing (ICAC)*, Jul. 2016, pp. 374–379. doi: 10.1109/ICAC.2016.29.
- [7] M. Grieves, “Digital Twin: Manufacturing Excellence through Virtual Factory Replication,” Mar. 2015.
- [8] “Industry 4.0 and the digital twin,” *Deloitte Insights*.  
<https://www2.deloitte.com/us/en/insights/focus/industry-4-0/digital-twin-technology-smart-factory.html> (accessed Apr. 14, 2021).
- [9] A. Fuller, Z. Fan, C. Day, and C. Barlow, “Digital Twin: Enabling Technologies, Challenges and Open Research,” *IEEE Access*, vol. 8, pp. 108952–108971, 2020, doi: 10.1109/ACCESS.2020.2998358.
- [10] M. Kaur, V. P. Mishra, and P. Maheshwari, “The Convergence of Digital Twin, IoT, and Machine Learning: Transforming Data into Action,” 2020, pp. 3–17. doi: 10.1007/978-3-030-18732-3\_1.
- [11] S. Mathupriya, S. Saira Banu, S. Sridhar, and B. Arthi, “Digital twin technology on IoT, industries & other smart environments: A survey,” *Mater. Today Proc.*, p. S2214785320389768, Dec. 2020, doi: 10.1016/j.matpr.2020.11.358.
- [12] “Why IoT is the Backbone for Digital Twin.”  
<https://www.ptc.com/en/blogs/corporate/iot-digital-twin> (accessed Apr. 10, 2021).
- [13] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, “Digital Twin in manufacturing: A categorical literature review and classification,” *IFAC-Pap.*, vol. 51, no. 11, pp. 1016–1022, Jan. 2018, doi: 10.1016/j.ifacol.2018.08.474.
- [14] R. Sahal, S. H. Alsamhi, J. G. Breslin, K. N. Brown, and M. I. Ali, “Digital Twins Collaboration for Automatic Erratic Operational Data Detection in Industry 4.0,” *Appl. Sci.*, vol. 11, no. 7, Art. no. 7, Jan. 2021, doi: 10.3390/app11073186.
- [15] T. Erol, A. F. Mendi, and D. Doğan, “Digital Transformation Revolution with Digital Twin Technology,” in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Oct. 2020, pp. 1–7. doi:

- 10.1109/ISMSIT50672.2020.9254288.
- [16] B. R. Barricelli, E. Casiraghi, and D. Fogli, “A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications,” *IEEE Access*, vol. 7, pp. 167653–167671, 2019, doi: 10.1109/ACCESS.2019.2953499.
- [17] “Altair Digital Twin Platform.”  
<https://www.altair.com/resource/altair-digital-twin-platform> (accessed Apr. 25, 2021).
- [18] C. C. MRICS, “How to make a digital twin: the options, types and outputs.”  
<https://info.vercator.com/blog/how-to-make-a-digital-twin-the-options-types-and-outputs> (accessed Apr. 12, 2021).
- [19] D. Burnett, J. Thorp, D. Richards, K. Gorkovenko, and D. Murray-Rust, “Digital twins as a resource for design research,” in *Proceedings of the 8th ACM International Symposium on Pervasive Displays*, New York, NY, USA, Jun. 2019, pp. 1–2. doi: 10.1145/3321335.3329685.
- [20] K. Borodulin, G. Radchenko, A. Shestakov, L. Sokolinsky, A. Tchernykh, and R. Prodan, “Towards Digital Twins Cloud Platform: Microservices and Computational Workflows to Rule a Smart Factory,” in *Proceedings of the 10th International Conference on Utility and Cloud Computing*, New York, NY, USA, Dec. 2017, pp. 209–210. doi: 10.1145/3147213.3149234.
- [21] “Digital Twins and AI: Transforming Industrial Operations.”  
</Read/31897/digital-twins-ai> (accessed Apr. 14, 2021).
- [22] N. Jazdi, B. Ashtari Talkhestani, B. Maschler, and M. Weyrich, “Realization of AI-enhanced industrial automation systems using intelligent Digital Twins,” *Procedia CIRP*, vol. 97, pp. 396–400, Jan. 2021, doi: 10.1016/j.procir.2020.05.257.
- [23] S. Y. Teng, M. Touš, W. Leong, H. Shen, H. Lam, and V. Máša, “Recent advances on industrial data-driven energy savings: Digital twins and infrastructures,” *Renew. Sustain. Energy Rev.*, vol. 135, p. 110208, Jan. 2021, doi: 10.1016/j.rser.2020.110208.
- [24] “What is a Digital Twin?,” *TIBCO Software*.  
<https://www.tibco.com/reference-center/what-is-a-digital-twin> (accessed Apr. 25, 2021).
- [25] “Living Heart Project | SIMULIA™ - Dassault Systèmes®.”  
<https://www.3ds.com/products-services/simulia/solutions/life-sciences/the-living-heart-project/> (accessed Apr. 25, 2021).
- [26] F. Tao *et al.*, “Digital twin-driven product design framework,” *Int. J. Prod. Res.*, vol. 57, no. 12, pp. 3935–3953, Jun. 2019, doi: 10.1080/00207543.2018.1443229.
- [27] X. Chen, E. Kang, S. Shiraishi, V. M. Preciado, and Z. Jiang, “Digital Behavioral Twins for Safe Connected Cars,” in *Proceedings of the 21th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*, Copenhagen Denmark, Oct. 2018, pp. 144–153. doi: 10.1145/3239372.3239401.
- [28] “Digital Factory: A Digital Manufacturing Lighthouse,” *siemens.com Global Website*.  
<https://new.siemens.com/global/en/company/stories/research-technologies/topics/digital-factory-siemens-electronic-works-chengdu.html> (accessed Apr. 23, 2021).
- [29] “Bentley Systems’ infrastructure digital twins to move to NVIDIA Omniverse,” *Mass Transit*, Apr. 12, 2021.  
<https://www.masstransitmag.com/technology/article/21218108/bentley-systems-infrastructure-digital-twins-to-move-to-nvidia-omniverse> (accessed Apr. 23, 2021).
- [30] G. Grinshpun, T. Cichon, D. Dipika, and J. Rossmann, “From Virtual Testbeds to Real Lightweight Robots: Development and deployment of control algorithms for soft robots,

- with particular reference to,” in *Proceedings of ISR 2016: 47th International Symposium on Robotics*, Jun. 2016, pp. 1–7.
- [31] F. Tao, M. Zhang, Y. Liu, and A. Y. C. Nee, “Digital twin driven prognostics and health management for complex equipment,” *CIRP Ann.*, vol. 67, no. 1, pp. 169–172, Jan. 2018, doi: 10.1016/j.cirp.2018.04.055.
- [32] “How Digital Twin Technology is Disrupting Healthcare,” *Plug and Play Tech Center*. <https://www.plugandplaytechcenter.com/resources/how-digital-twins-technology-disrupting-healthcare/> (accessed Apr. 24, 2021).
- [33] “Digital twins: The future of healthcare delivery and improved patient experience.” <https://www.beckershospitalreview.com/healthcare-information-technology/digital-twins-the-future-of-healthcare-delivery-and-improved-patient-experience.html> (accessed Apr. 24, 2021).
- [34] “Disrupting The Healthcare Industry Using Digital Twin Technology,” *Challenge Advisory*. <https://www.challenge.org/insights/digital-twin-in-healthcare/> (accessed Apr. 24, 2021).
- [35] “Gazebo.” <http://gazebo.org/> (accessed Apr. 14, 2021).
- [36] “Gazebo Simulation | PX4 User Guide.” <https://docs.px4.io/master/en/simulation/gazebo.html> (accessed Apr. 14, 2021).
- [37] “Gazebo simulator,” *Wikipedia*. Dec. 12, 2020. Accessed: Apr. 14, 2021. [Online]. Available: [https://en.wikipedia.org/w/index.php?title=Gazebo\\_simulator&oldid=993767160](https://en.wikipedia.org/w/index.php?title=Gazebo_simulator&oldid=993767160)
- [38] L. Nogueira, “Comparative Analysis Between Gazebo and V-REP Robotic Simulators,” Dec. 2014. doi: 10.13140/RG.2.2.18282.36808.
- [39] N. Kousi, C. Gkournelos, S. Aivaliotis, C. Giannoulis, G. Michalos, and S. Makris, “Digital twin for adaptation of robots’ behavior in flexible robotic assembly lines,” *Procedia Manuf.*, vol. 28, pp. 121–126, Jan. 2019, doi: 10.1016/j.promfg.2018.12.020.
- [40] “CoppeliaSim Features - Coppelia Robotics.” <https://www.coppeliarobotics.com/features> (accessed Mar. 22, 2021).
- [41] “Using V-REP as a digital twin for real robotic inspections - Forum.” <https://forum.coppeliarobotics.com/viewtopic.php?t=7880> (accessed Apr. 23, 2021).
- [42] “Robotics simulation in Unity is as easy as 1, 2, 3! 「1、2 の 3 !」で簡単に出来る、Unityでのロボティクスシミュレーション - Unity Technologies Blog,” Nov. 19, 2020. <https://blogs.unity3d.com/2020/11/19/robotics-simulation-in-unity-is-as-easy-as-1-2-3/> (accessed Apr. 25, 2021).
- [43] A. Konrad, “Simulation of Mobile Robots with Unity and ROS,” p. 69.
- [44] U. Technologies, “Robotic simulation | Unity.” <https://unity.com/solutions/automotive-transportation-manufacturing/robotics> (accessed Apr. 14, 2021).
- [45] L. Pérez, E. Diez, R. Usamentiaga, and D. F. García, “Industrial robot control and operator training using virtual reality interfaces,” *Comput. Ind.*, vol. 109, pp. 114–120, Aug. 2019, doi: 10.1016/j.compind.2019.05.001.
- [46] “Point cloud - Wikipedia.” [https://en.wikipedia.org/wiki/Point\\_cloud](https://en.wikipedia.org/wiki/Point_cloud) (accessed May 18, 2021).
- [47] N. Rottmann, N. Studt, F. Ernst, and E. Rueckert, “ROS-Mobile: An Android application for the Robot Operating System,” *ArXiv201102781 Cs*, Nov. 2020, Accessed: May 16, 2021. [Online]. Available: <http://arxiv.org/abs/2011.02781>
- [48] “rosbag - ROS Wiki.” <http://wiki.ros.org/rosbag> (accessed May 16, 2021).

- [49] “CloudCompare - Open Source project.” <https://www.cloudcompare.org/> (accessed May 16, 2021).
- [50] “ROBOTIS e-Manual,” *ROBOTIS e-Manual*.  
<https://emanual.robotis.com/docs/en/platform/turtlebot3/features/> (accessed May 03, 2021).
- [51] “Documentation - ROS Wiki.” <http://wiki.ros.org/> (accessed May 16, 2021).
- [52] “Revit Software | Get Prices & Buy Official Revit 2022 | Autodesk.”  
<https://www.autodesk.com/products/revit/overview> (accessed May 16, 2021).

## Extras

### Annex 1

For the creation of the digital twin in Gazebo, the GitHub platform was used. The GitHub repository contains the code developed as part of this project. The entire folder contains all the codes that were written in this project.

[https://github.com/ut-ims-robotics/unitartu\\_tech\\_gazebo/](https://github.com/ut-ims-robotics/unitartu_tech_gazebo/)

### Annex 2

The GitHub repository below includes major packages for the Jackal robot. The *jackal\_navigation* subfolder handles the map development process.

<https://github.com/jackal/jackal.git>

### Annex 3

The links below show the step-by-step procedure in developing a Gazebo model.

[http://gazebosim.org/tutorials?tut=build\\_model](http://gazebosim.org/tutorials?tut=build_model)

To import mesh, [http://gazebosim.org/tutorials/?tut=import\\_mesh](http://gazebosim.org/tutorials/?tut=import_mesh).

### Annex 4

The following XML lines are needed to define a custom Gazebo model path. These lines of code were added to the *package.xml* file in the *unitartu\_tech\_gazebo* package.

```
<depend>gazebo_ros</depend>
<!-- The export tag contains other, unspecified, tags -->
<export>
  <!-- Add GAZEBO_MODEL_PATH -->
  <gazebo_ros gazebo_model_path="${prefix}/models/" />
</export>
```

## **Non-exclusive Licence to Reproduce Thesis and Make Thesis Public**

I, Kwasi Akuamoah Boateng

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, Digital Twin of a Teaching and Learning Robotics Lab, supervised by Associate Professor Karl Kruusamäe.
2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.
3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.
4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Kwasi Akuamoah Boateng

20/05/2021