Alma Mater Studiorum – Università di Bologna

DOTTORATO DI RICERCA IN

Meccanica e Scienze Avanzate dell'Ingegneria

Ciclo XXXII

Settore Concorsuale: 09/A3 – PROGETTAZIONE INDUSTRIALE, COSTRUZIONI MECCANICHE E METALLURGIA

Settore Scientifico Disciplinare: ING-IND/15 - DISEGNO E METODI DELL'INGEGNERIA INDUSTRIALE

Direct modeling techniques in the conceptual design stage in immersive environments for DfA&D

Presentata da: Francesco Osti

Coordinatore Dottorato

Prof. Marco Carricato

Supervisore

Prof. Alfredo Liverani

Esame finale anno 2020

ALMA MATER STUDIORUM - UNIVERSITY OF BOLOGNA

DOCTORAL THESIS

Direct modeling techniques in the conceptual design stage in immersive environments for DfA & D

Author:
Francesco OSTI

Supervisor:
Prof. Alfredo LIVERANI

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Design and Methods in Industrial Engineering
Department of Industrial Engineering (DIN)

October 31, 2019

Declaration of Authorship

I, Francesco OSTI, declare that this thesis titled, "Direct modeling techniques in the conceptual design stage in immersive environments for DfA & D" and the work presented in it are my own. I hereby certify that the thesis I am submitting is entirely my own original work except where otherwise indicated. I am aware of the University's regulations concerning plagiarism, including those regulations concerning disciplinary actions that may result from plagiarism. Any use of the works of any other author, in any form, is properly acknowledged at their point of use.

Signed: Francesco Osti

Date: **October 31st, 2019**

ALMA MATER STUDIORUM - UNIVERSITY OF BOLOGNA

Abstract

School of Engineering
Department of Industrial Engineering (DIN)

Doctor of Philosophy

Direct modeling techniques in the conceptual design stage in immersive environments for DfA & D

by Francesco OSTI

Due to the fast – growing competition of the mass – products markets, companies are looking for new technologies to maximize productivity and minimize time and costs, while maintaining high product quality. In recent years, immersive technologies have moved from labs and research environments to industrial contexts, looking for practical use in different areas of design and manufacturing. In the perspective of Computer Aided Process Planning (CAPP), companies want to optimize fixture design and assembly planning for different goals. To meet these demands, the designers' interest in Design for Assembly and Disassembly is growing considerably and is increasingly being integrated into the CAPP. The use of immersive technologies for the fixture design and the assembly and disassembly evaluation is a trend of research but a direct modeling system in an immersive environment has not yet been proposed. The work described in this thesis aims to exploit immersive technologies to support the design of mating elements and assembly / disassembly, by developing a data exchange flow between the immersive environment and the modeling environment that provides the high – level modeling rules, both for modeling features and for assembly relationships. The main objective of the research is to develop the capability to model and execute simple coupling commands in a virtual environment by using fast direct modeling commands. With this tool the designer can model the coupling elements, position them and modify their layout. Thanks to the physical engine embedded in the scene editor software, it is possible to take into consideration physical laws such as gravity and collision between elements. A library of predefined assembly features has been developed through the use of an external modeling engine and put into communication with the immersive interaction environment. Subsequently, the research involved the study of immersive technologies for workforce development and training of workers. The research on immersive training involved industrial case studies, such as the projection of the disassembly sequence of an industrial product on a head mounted display, and less industrial case studies, such as the manual skills development of carpenters for AEC sectors and the surgeon training in the pre – operative planning in medical field.

Contents

List of Figures

Chapter 1

Introduction

1.1 General overview of current CAD modeling

During the last twenty years, the market of commercial high end Computer Aided Design (CAD) software has seen the unrivaled dominance of parametric history – based modeling. First of all, parametric modeling has its strengths in the dimension driven approach and in its highly automated workflow. Moreover, the feature – based modeling technique, derived from CNC machining, constitutes a reasonably easy way for designers to conceive and mentally visualize product geometry. On the other hand, some drawbacks are associated to these strengths, such an example the demand for pre – planning the entire modeling and the extreme rigidness to rapidly adapt to changes in product requirements during the early stages of product development. Parametric modeling shows specific weaknesses when it deals with modeling in the conceptual stage. During concept phase, design intent is used to change drastically and rapidly and parametric modeling is unsuitable to follow the design intent mutations, when designers need to try and redesign several times before finding the right solution. When the design process is finished and the solution is finalized, designers restart the modeling work in order to create a robust model, able to harness design intent from the beginning. This workflow doubles the work of designers. The main advantage of this workflow is that, eventually, final design intent is well preserved. However, modeling in a robust manner to fix design intent in the product 3D model requires experience and hard training. Firstly, experience because designers need to change mindset from engineers to programmers. History – based parametric modeling has been initially developed to meet hardware specs from the 1980s: CAD developers had to divide the amount of geometric information of a product into smaller computable parts, hence, from assemblies into single parts, and features into 2D sketches. For this reason, the feature tree can be seen as a piece of code that can be run iteratively and capable of creating a model according to the initial set of parameters. Secondly, hard training because designers must set up a well – thought modeling strategy to build a 3D model robust and resilient to changes. For this reason, many companies devote resources in training designers in order to standardize modeling strategies. Thanks to the training, the company ensures that design intent is preserved for a long time and it becomes a company know – how. On the other side, direct or explicit modeling techniques are not new in the CAD world. The first commercial application has been introduced by Siemens PLM in the late 2000's under the name of Synchronous Technology which was presented as a more intuitive way to edit and interact with parametric models. However, this type of modeling has always been seen as a weaker 3D modeling, unable to compete with the power of parametric modeling. The great advantage of direct modeling is its capability to easily adapt to design changes. In the last years, direct modeling has seen a great development, especially thanks to the advent of new design methodologies, such as generative design, and design tools, such as topology optimization techniques, and manufacturing technologies, such as additive manufacturing, that have given to the designer the freedom to create new shapes for products capable to maximize the fulfillment of product requirements. For example, it is extremely difficult and cumbersome the modeling of structural parts that have been topologically optimized with parametric modeling techniques. For the same reason, the manufacturing of components developed through generative design techniques is irreconcilable with CNC machining. For these reasons, direct modeling and its flexibility is a suitable solution for innovative design approaches, but at the same time future explicit 3D modelers should be able to use or introduce some rule – based approach in its architecture. Direct modeling has been widely studied from single component perspective. Research works on direct modeling procedures on a single component range from coding of new direct modeling commands and development of features recognition algorithms, to conversion techniques of direct modeling features into parametric history – based modeling sequences. Few researches about direct modeling have been carried out on multi – components or assemblies. For this reason, our research work concentrated on explicit modeling of assemblies, with a particular focus on direct modeling of assemblies in immersive environments. The association of direct modeling with immersive technologies is a natural consequence since immersive technologies, like direct modeling, has begun to be largely employed in the early design stage. it is no coincidence that in recent years virtual and augmented reality applications have been developed for conceptual 3D modeling and design review. Some of these applications are still in the R&D phase or labs. Others, such as Mindesk, Iris VR, Marui, are now commercial plug – in for several CAD software. Furthermore, PTC has acquired Vuforia Engine and developed its own scene editor, Vuforia studio, as a proof that CAD providers and immersive technologies are going to be interconnected tools of the product development process. In this thesis, we describe the research on the implementation of direct modeling techniques in the game engine Unity3D for the modeling and simulation of assembly and disassembly. In the Chapter 1, we present a comprehensive State of Art of direct modeling implementations over the last years. In Chapter 2, we illustrate Materials and Methods used in order to carry out the research. The results of the direct modeling of assembly constraints are outlined in Chapter 3. The applications of the assembly and disassembly simulations are reported in Chapter 4 and Chapter 5. Chapter 6 depicts future trends of the research.

1.2 General considerations on direct modeling

When speaking of direct modeling and history – based parametric modeling is important understand what the designer is trying to do while modeling with these two techniques. There are four possible situations. The first two situations are designing product and documenting those products and exists a clear distinction between those two operations. Documenting products consists in creating engineering documentation for downstream applications, like manufacturing, quality checking, service and maintenance, etc. Designing products is about exploring and iterating on design so it can meet the requirements of the product. The third situation is the simulation of products' performances and this step consists in abstracting and simplifying a design in preparation for simulation. Lastly, producing the product which requires the creation of downstream tooling, like tool and die, and numerical control code to manufacture the product. We can say that in designing product and simulating its performance, designers need a very simple model, in order to edit rapidly and to maintain simulation time as low as possible. Hence, in these two phases, direct modeling is the best suitable solution because the changes and the simulations can be performed quickly. On the other hand, when the designer is requested to prepare product documentation or to design manufacturing tool for producing the product, a fully detailed model of the product is necessary and in this case parametric history – based modeling is the best suitable solution in order to ensure the highest manufacturing standards, automation and repeatability. Even assembly and disassembly planning falls in the producing situation; hence assembly constraints are modeled by parametric history – based modeling tools rather than with direct modeling techniques. It is also important to understand the difference in creating geometry between parametric history – based modeling and direct modeling. With parametric history – based modeling, the user creates 3D geometry progressively with features. Features range from simple sketch – based features, like in extrusions and revolutions, and medium complexity edge – based features, such as chamfers and rounding, up to very complex features like sweeps and blends. The operations that are used to create a specific geometry are remembered, so the user has the history of how geometry is built and created. This also creates interdependencies between features and, more important, between components in assemblies because assembly features, or assembly constraints, rely on mating surfaces that are generated by sketch – based features in the two components. Hence, editing the feature tree can lead to a failure in the regeneration both of the single component model or, worst, of the entire assembly. On the other hand, through direct modeling geometry can be created by extrusion of 2D sketches or by instancing primitive geometries, hence, in a similar way to features, but the history is not retained. This means that manipulation and selection of geometry is not constrained by how the geometry was created. The user can include parametric modifications but the history of those modifications is not memorized. It is also worth noting that these considerations regard not only 3D geometry, but also 2D geometry. Sketching tools let the user draws entity by entity the 2D section of a component and several 2D constraints, like parallelism, perpendicularity, concentricity, enforce properties between instanced entities and are applied constantly during changes, like in parametric history based modeling. On the other side, drafting tools are more like direct modeling: user can create the entities and the properties of the entities are applied once at the creation, but are not persisted during changes. In a recent survey [1] by PTC regarding the use of CAD tool in the conceptual design phase, 92% of respondents think that design process would benefit by exploring design alternatives during concept design. There is a demand for speed and flexibility while evaluating those alternatives because designers are not caring about history in the concept phase but at the same time, only 16% of respondents use a 3D modeling software in this early phase of product design and that is because direct modeling tool has not gained enough authority and maturity at designers' eyes. Design team feel to waste time in using 3D modeling tools in the conceptual design phase because they are not able reuse any model they create in this phase in the downstream steps of product design. In fact, 61% say that drawings from the concept phase need to be created again to help and support the subsequent detailed engineering phase. Another problem is that 68% of respondents say that a concept solutions start and evolve from a pre – existing design but, since they are not able to reuse data, designers are bound to start from scratch, duplicating modeling efforts. They call this design repurposing. The most common way to perform design repurposing is to use 3D direct modeling commands. It is worth noting that in design repurposing they do not want to completely delete old geometry or cancel original design intent. They rather want to bend original geometry and original design intent to new functions in order to verify the suitability of already modeled geometries to newer product requirements. This process, if performed, would speed up the iteration time of concept phase and drastically cut the cost of product design development. .

1.3 General considerations on immersive technologies

One of the newer tool to verify if the product meets the requirements is represented by immersive technologies: virtual reality and augmented reality. Thanks to these technologies, designers share interactive 3D experiences across the company and suppliers. Or they can perform design review in collaborative virtual environments in order to detect design mistake earlier. Eventually, designers can develop easier assembly and disassembly sequences and train operators in these operations in immersive environments without expensive prototypes. VR and AR technologies allows designers to communicate concepts in a 3D visual way, and most of the time, in real – life scale. This allows them to explore and share design ideas in a more comprehensive way, from different point of views and in multiple scenarios. In some field, such as the automotive field, AR and VR are the most intuitive way to evaluate multiple configurations, for example in the choice of the materials and textures for car interiors. The benefits of immersive technologies do not end in the conceptual design phase. Data visualization, for example data coming from Finish Element Analysis, has been widely used in engineering department for a long time to visualize mathematical results and better understand components behavior. Now engineers can exploit the 3D product model to analyze the consequences of engineering decisions. To support engineers, new tools and functionalities should be developed in order engineers to measure and simulate real world phenomena in immersive environments. For this reason, the efforts of game engine suppliers aim to develop always better physical engine in order to better replicate physical laws and simulate physical phenomena. For example, a simulation engineer can better explain the results of a Computational Fluids Dynamics simulation by using the real – life scale 3D model and virtual smoke. In other words, immersive technologies help engineers to comprehend and communicate data and decisions in a more perceivable way. The power of immersive technology impacts also the manufacturing engineering for plant design. Manufacturing engineers can take advantages from 3D plant and manufacturing line design in order to investigate different plant layouts by immersing themselves in a virtual plant. Even if the virtual model of the plant is not available, manufacturing engineers can use point clouds from 3D scanners to augment existing equipment with the new line. In this context, immersive 3D can be employed to better perceive the role of workers in the factory. Such an example, ergonomics of working cells can be design and evaluated in VR or AR. While in VR, operators can be trained in collaborative working with robots. Lastly, VR and AR can reduce the costs and increment the learning rate of training courses. Beyond the so – called "wow effect" of giving high – tech training to operators, research in the field of VR training is evaluating the benefits of immersive training in the industrial context. In fact, training is one of the first application for companies that want to implement VR in their factories. VR training cannot completely replace traditional training but VR training research field aims to demonstrate that the more the VR training is performed the less the traditional training is required. For companies, this means that they do not need to keep manufacturing equipment offline or to sustain for workers travel costs to simulator facilities or other manufacturing sites. Moreover, decreasing traditional training means that workers can be trained on virtual manufacturing lines before the real line is set up, guaranteeing that operators are productive as soon as the plant is operational. VR training has also safety benefits since it reduces the risks of injuries and accidents when workers has to handle hazardous materials of harmful equipment. Finally, AR training with the use of haptic supports or physical hardware can enhance the training effectiveness and develop specific muscle memory in order to automate specific movements.

1.4 Research objective

Immersive technologies are considered a natural continuation of the CAD modeling environment. The user can view and interact with virtual models and can modify the models within the virtual environment. One of the applications of immersive technologies in industrial contexts is represented by virtual prototyping. Virtual prototyping stems from the idea of replacing expensive physical prototypes with digital mock-ups in immersive virtual environments. The digital mock - up does not have the mere function of displaying the product, but embeds physical behaviors and properties that enabled the digital mock - up to act like the real product to external solicitations or interactions with the user. In all these applications, the CAD models are modeled with the 3D modeling software used by the company. Later, parametric CAD models are converted into polygonal models, i.e. meshes, for importing them into the virtual scene modeling software. In the scene editor environment are then added features, properties and behaviors to the model in order to be able to interact and simulate the use and behavior of the model. This approach has some limitations:

- The modeling software used by companies is mainly parametric CAD. Hence, exact dimensions are required to shape the piece. This data is not usually known with precision in the conceptual design phase.
- The topological relationships and the assembly constraints introduced in the parametric modeling, both at the model level and at the assembly level, are lost with the conversion into a polygonal model for importing into the software scene editor.
- In order to make changes and visualize those changes in the immersive environment, it necessary to edit the model in the CAD software and import the model back into the scene editor software.

These limitations can be overcome by integrating the CAD functions into the immersive environment. In this way all the modeling activities could be carried out within the VR environment, through intuitive and direct operations and without the cumbersome data conversion process between CAD and scene editor software. CAD integration - VR system is not a simple goal. VR systems have limited capabilities for 3D modeling, both for modeling individual components, and for modeling assemblies with related constraints between components. This is due to the fact that CAD software uses solid and extremely complex surface modeling methods for a mathematically exact model definition. On the other hand, the polygonal representation used in the immersive environment is a finite resolution model representation in order to allow model visualization and user interaction on – the – fly. Moreover, the polygonal mesh does not embed topological data or assembly constraints. Hence is not suitable for precise CAD modeling.

The research described in this thesis aims to develop a hybrid – modeling environment, an optimized combination of direct modeling techniques with embedded constraints high - level rules system, in order to boost the preliminary phase of design. The hybrid - modeling environment will be a support tool for the designer exploring the design space and its quick evaluation. The set of high-level rules in the immersive environment allows the direct modeling of assembly features and assembly constraints. The system is based on a three level architecture. Firstly, a low level polygonal mesh representation in the immersive environment. Secondly, the data exchange module that converts the input used to create polygonal geometries into readable data for the high level rules system. Lastly, the high level system of rules for topology relationships used to define assembly components and mating elements, and for assembly constraint definition between those mating geometric elements. Thanks to the immersive assembly modeling system, the user is able of modeling via direct, or explicit, techniques new geometries on an imported model, or to quickly edit existent geometry of the model. The creation of new geometry and the modification of old geometry are intended to create new assemblies in immersive environments. For this reason, we focused on the geometry that defines the joints features between components: face to face, pin/hole joints, hinges or prismatic coupling are the type of joints that has been considered. All these type of joints are simple to model in the conceptual design phase because they are composed by simple geometrical element like planar surfaces, for face to face or prismatic joints, or cylindrical surfaces, for pin/hole joints, or even simpler elements, like single edges in hinge joints. In the conceptual design phase the objective is to model these joints to quickly evaluate their feasibility. The integration of the immersive hybrid - modeling environment will be used as an innovative Computer Aided Engineering (CAE) to simulate and evaluate manufacturing processes, with a specific focus on assembly and disassembly procedures. The research aims to simulate and test the assembly or disassembly that derives from rapidly modeled coupling elements, in order to evaluate the effectiveness of the assembly and disassembly sequence in a Design for Assembly and Design for Disassembly approach. Lastly, we aim to use the immersive system to train operators in those sequences. Our final goal is to assess the validity and the effectiveness of immersive technologies in multiple scenarios: starting from more industrial contexts, where immersive technologies have been widely used and tested during the last years, up to Architecture, Engineering and Construction (AEC) sector where the use of these technologies is recent and not deeply implemented.

Chapter 2

State of Art

2.1 State of Art of Direct Modeling Techniques

Digital drawings and 3D models have completely replaced paper drawings in the data exchange workflow among the stakeholders who participate to the product development process. Various types of information merge into what is called product data [2]: 3D models, product briefs, data tables, etc. All this information is now managed at the company level by Product Data Management (PDM). The PDM is a software tool, employed in concurrent engineering environments, that allows you to increase efficiencies within the development processes of existing products, thus improving the management of product design data. The basic product data is represented by the 3D model of the product in the native CAD format with which it was modeled, the interoperability of 3D CAD models is fundamental in concurrent engineering contexts where each phase of product development requires a different file format [3]. Hence, the types of representations for 3D models is crucial for a quick and seamless product development process. According to the [4] the 3D CAD model is a mathematical representation of the product. This representation is used to describe and foresee the behavior of the product upon specific requests of the designer. In the history of mechanical CAD software, two types of approaches, or languages, to develop 3D model representations, can be enunciated: declarative approach and procedural approach. In the declarative approach, the representation is declarative, or explicit, meaning that the geometry is instantly available, without the need for the software to operate calculations or operations [5]. On the other hand, in the procedural approach or history – based or generative approach, procedural representations are characterized by a sequence of operations that, only when have been evaluated, allow the representation of the geometry. Then, it exists a hybrid type of representation which merges elements of the explicit representation with elements of the procedural representation [6]. Nowadays, in the market of commercial CAD software, the explicit approach is represented by the Boundary Representation (B – Rep) based software, while in the procedural approach are listed the history – based parametric and feature – based CAD modelers. This classification is not completely accepted by researchers since featured – based software embeds B – rep elements. In fact, even in parametric history – based software, geometric elements are connected and employed in order to represent faces, edges and vertices of the contour, or Boundary, of a solid [7]. In the B – Rep representations, two set of data exist: geometric information and topological information. Geometric information describes the shape and relative positions of geometric elements; while topological information describes the relationships between the elements [8]. The boundary of the solid is limited by geometric elements which define patches of surfaces. The topological relationships are applied to each patch in order to described the solid, for example, in a cube the perpendicularity between two patches convergent on the same edge, or the parallelism between two opposite patches. However, the patches are unsuitable for rendering in a computer screen for interactive visualization or for behavioral simulation, due to their mathematical descriptions and due to the fact that they require extremely high computational efforts [9]. For this reason, the mathematical geometry described via B – Rep needs to be converted into a simpler representation which approximates the exact geometry of the 3D model. The solution is the discretization of the surface into a mesh of smaller elements, usually triangles, which approximates the mathematical description of the surfaces that limit the solid [10]. The approximated mesh model is mandatory for all the downstream applications of the product development process in order to simulate, in reasonable time, the behavior of the product. Classical simulations performed to predict product behavior are the Finish Element Analysis or Computational Fluid Dynamics simulations. However, in recent years, another type of simulation has arisen due to the joint advent of immersive technologies and innovative design approaches, such as Design for Assembly (DfA) and Design for Disassembly (DfD). These simulations consist of the evaluation and the simulation of assembly and disassembly sequences to meet several optimization objectives. The optimization objective function can comprehend time and cost saving aims, ergonomics aspects or environmental impact considerations. Besides traditional simulations and optimizations of assembly and disassembly operations performed via mathematical algorithms, recent immersive technologies offer a more interactive approach and give the designer a more comprehensive insight of the issues related to assembly and disassembly, since these operations require the interaction with the product. Assembly and disassembly sequence simulation performed with immersive technologies requires specific tools and software. One of the software used to create immersive content and interact with immersive technologies is the game engine Unity 3D, and, as any other type of simulations, it relies on mesh models. However, since DfA and DfD are design approaches implemented in the early design stage, the designer should also have the capabilities to edit and create new geometries with direct modeling techniques inside Unity, not only to interactively simulate the sequence. Among the direct modeling commands implemented in several CAD systems, push and pull technique represents a very simple way to edit the 3D model. Push and pull commands were originally called tweaking and the first description of this technique can be found in [11]. In this work the authors described the need for the designer to add operations of local influence, or action, capable of modifying already – built geometries. The new tweaking operations included the edits of the geometry of a face, with consequently modification of geometrical and topological relationships between faces, and the movement of holes within a face. The authors described also some problems coming from the use of these direct modeling tweaking commands. For example, the creation of non – manifold geometries, as a consequence of the overlapping of the new topological relationships with pre – existent topology constraints of patches and faces which had not been edited. More recently, push and pull edits have been studied by [12]: in this work, the features of a 3D model can be moved and rotated by simply dragging the feature, but deeper changes are not supported, for example changing the profile of a section. In the same way, the authors of [13] described a method to change position and rotations to selected features but they did not rely with substantial edits in the shape of the features to be moved. [14] proposed a method to apply direct modeling edits to parametric models and to reorganized the features tree in order to convert direct edits in parametric features, but the topology of the features to be moved should remain unedited. All these example are part of the research trend that conceive push and pull operation of a single face, as operation to be performed on the volume that contains that face. This approach is robust and reliable but its application is limited because the user is merely changing the volume containing the face, i.e. changing the position of the 2D sketch or the 2D profile, rather than actually changing the topology of the volume containing the face. In conclusion, we can say that these approach aims to convert push and pull edits into parametric features edits and this approach is not suitable for B – Rep models which have no features tree to rely on. The work by [15] addresses the problem of direct editing a polygonal mesh and their research is similar to the research presented in this work since both of them are about editing of meshes. However, the research by [15] is restricted to meshes derived by the discretization of 3D models composed by planar surfaces. Hence, it is a limited perspective in the context of mechanical components modeling. Subdivision surfacing is a novel modeling technique which can be seen both as surface modeling or as mesh editing. With subdivision method the polygonal faces of a mesh are refined by subdividing each face into smaller faces. Every face is then edited by operating on the control points of the grids and lastly the surface interpolating all the control points is created with piecewise patches and it was firstly used to reconstruct 3D models from 3D scanned data [16]. As stated by the authors, the subdivision method is especially useful for complex shape products, or free – form products, because mechanical components can be usually divided in simpler geometric entities, like parallelepipeds, cylinders and spheres. For such objects, segmentation techniques and shape recognition algorithm are more suitable. Moreover, the subdivision method control points grid is particularly difficult to be handled by novice designers and cumbersome for creating specific features like sharp edges, which are common in mechanical components. Furthermore, it is computationally heavy due to the mathematics underneath. Hence, the subdivision modeling is not suitable for direct modeling of mechanical components, especially in real time immersive environments.

2.2 State of Art of Assembly Modeling

The two approaches, procedural and explicit, appears even in the assembly modeling. The first to study the assemblies from a procedural and a declarative perspective were [17]. In their work, the authors studied the methods for determine the position of an assembly component from high – level relationships, or constraints, both in a procedural and in a declarative approach. In the procedural approach, the constraints definition is checked as the assembly is assembled. As stated by the authors, this method works only in a Bottom – up assembly approach, i.e. all the parts have already been modeled, because assembly constraint calculations start from and depends on the feature modeler which gives dimensions of the feature, shapes of the feature, and feature – face relationships, and other parametric data. In the declarative approach, the designer set relationships between geometric entities of each assembly feature. This second approach has the advantages that: it separates the constraints definition from the constraint validation, while in the procedural approach these two operations are linked, and secondly, that the assembly modeling system is independent from the geometric engine of the modeler. In [18] the authors proposed the assembly design following a top – down approach. Thanks to the top – down approach, the designer can identify key characteristics of an assembly that ensure to meet customer requirements. in this work, key characteristic are the mating surfaces and the authors described two types of key characteristics: type

1 are the key characteristics that allow to perfectly position one component with respect to the other, for example the two flipped edges where fastening are riveted; type 2 are the key characteristics that do not fix the position of the two components. In [19] the authors demonstrated that assembly features are as useful for assembly planning as the features based modeling is useful for manufacturing. Moreover, the relationships between components can be the basis for product modeling since the coupling elements can progressively add more details, until the entire product, with all the sub – components, is fully defined. Another work on assembly co – design is the one by [20]. In this work the authors proposed a modification system and an edit propagation workflow based on Extensible Markup Language (XML) between the different clients involved in the design process. In order to manage design modification propagation and perform real – time changes on the model, two type of XML file are written at the moment of the modeling stage. First XML file contains the parameters of each feature of the 3D model. The second XML file contains the information containing assembly features. The two files are then sent from one client to the other. This method is similar to ours, because our method implements a client – server system and a data exchange in a text format. Later, in [21] the authors proposed an open source assembly model called Open Assembly Model (OAM). The proposed model includes data of the assembly as a concept model, i.e. as a self-contained product, and as a data structure, i.e. containing relationship and dependent information between each component. This information is contained in a single custom class developed via the API of the CAD modeler they used. For this purpose, in particular, the authors implemented the data structure of STEP format. Being open source allows OAM to be seamlessly integrated in different CAD software. Moreover, this is one of the first research work where it is explicitly declared the use of assembly process planning and virtual assembly. They wanted to apply their method inside the Virtual Assembly Design Environment (VADE) [22]. However, no papers on the application of this method in a virtual reality environment has been published. In [23] the authors developed a new assembly feature class using the UG/Open Application Programming Interface (API). In particular, their method developed an Associative Assembly Feature (AAF) capable of creating several associations: an association between parts not already geometrically defined; an association between geometric entities that defines a mating interface; and an association between component geometry and an intermediate geometry in order to define a new part. In the example they provide, however, their interface looks like more a configurator of different products, based on some parameters, rather than a tool for assembly edits and changes. In [24] the authors proposed a API class development based on the work by [21]. Also in this work, the authors used the STEP language to model the assembly but they split the information originally contained in the single class of OAM and they set a two separate classes: one for assembly parameters and one for assembly constraints. For this reason, they called this method OAM+. Moreover, the OAM+ classes enable to embed more data of different nature. For example, they managed to integrate load data in the assembly, meaning that the assembly model can be used for simulation and Finish Element Analysis. The use of STEP standard allows this method to be integrated in several CAD modelers, as with it was for OAM method. From the concept of Assembly Feature presented in the previous research work, the work by [25] introduces the concept of Virtual Assembly Feature (VAF). As stated by the authors, VAF embeds a function – oriented approach and has a more engineering foundation. In their work, VAF becomes the tool for assembly sequence planning and assembly feasibility evaluation in virtual environments. In VAF there are four types of information: geometry information

about the parts; mating information regarding the constraints between parts; operation and movements required to assemble; and generic auxiliary information for the assembly task. When a virtual assembly task is performed, all this information is recollected and ensures that the assembly planning has engineering validity. By using different RGB value for each surface of the parametric model, they managed to develop an assembly workflow for meshes model in virtual environments. In recent years, digital prototyping has given the designers the capability to study and explore virtual assemblies with simulations. This aspect has been studied by [26]. The authors have developed an Assembly Feature Pair (AFP) capable of modeling and simulating assemblies. Thanks to AFP embedded into the part model, designer can introduce in the single component model the behavior of the component when interacting with other components. In this way, the work for creating the assembly is switched in the work for instancing the AFP in the part model. The advantage to embed AFP in components models is that components acquire knowledge of what they will be able to do in the future, i.e. how they will be assembled, without using a separate assembly model. This has also a drawback, because, as state by the author, the foreseeing capabilities could work only if a well-established library of AFP exists: a component can foresee how to self – assemble only if every other component of the final product has the AFP integrated in its model. However, as we can see from the literature review, assembly modeling is a branch of the feature – based modeling, and even using STEP language, recognizing features on the face and surfaces of the model is required in order to assemble the model of the final product.

2.3 State of Art of Industrial Immersive Application for Assembly Modeling

Besides the work proposed by [21], one of the first example of constraint definition in VR environment is represented by the work in [27] in which the authors developed a modeling technique that leverages the "allowable movements" of the 3D model. The "allowed motions" enable the accurate pose of the model and the automatic constraint of the model. The approach is based on an assembly constraints recognition workflow. These relationships are stored in a graph and the allowed motions direction are computed starting from the graph using the method proposed by [28]. The proposed system was able to automatically recognize face to face constraint, cylindrical fit and concentric constraints. All these joints could be performed in the 3D VR environment. Moreover, the relationships graph could be used to propagate the assembly relationships among all the other parts of the assembly. However, the system could handle only these simple constraints, and more complex assembly constraints required further research. The use of allowed movable directions is an interesting approach and it had some development, especially in the construction of the collision matrix. The great drawback of this method is that the 3D model still required to be edited in the CAD software and imported in the scene editor, if changes were required. Later, in the work by [29] proposed a method based on grid snapping, as the method implemented by our system. The input for the geometry creation come from a 3D mouse, as in our system the input for model rotation and movement come from the immersive input device. In the proposed method the modeling in virtual environment is solid and effective but the constraints definition is based upon a set of predefined rules which are too rigid for an industrial use. In the research by [30] the authors developed a method for solid modeling in a semi – immersive VR environment. The method is based on the modification of control points of the primitive shape. The user can drag the control points and edit the geometry instead of inserting the parameters by keyboard. In the work described in [31] the authors use three methods to model and constrain objects one to each other. The simplest way to place object is by mean of a sliding bar that allows the rotation and the translation of the components. This method is used for independent or absolute positioning. The second method is by constraint – based manipulations which is used for specific modeling operations, such as adding block or creating slots or pockets. The third method is by applying a set of constraints to two components and position the second selected object with respect to the first one. This is the method used for creating assembly in a virtual environment. However, the research did not deal with the management of assembly constraints and the possible constraints conflicts the user can generate. The problem of precise positioning of components of an assembly within virtual environments has been tackled by [32]. In their work, the author described a system able to recognize assembly relationships and to create constrained paths of movements based on the recognized relationships. The system is developed in a CAVE environment and tested in three case studies with two group of students. Thanks to the feature recognition process implemented in the system, the authors managed to reduce the errors of assembling two components in a way not conforming to the design intent, for example inserting a bolt in a pin with a different diameter. Regarding the errors in assembly procedures, a large research has been conducted. In [33] proposes the extraction of the assembly/disassembly sequence from an expert user performing the task in VR environment. A data mining process is applied to discover assembly/disassembly models and generate the proper sequence. Finally, a new user can learn and practice the assigned task. The assembly components are Lego - like boxes. 20 users were involved performing the learning stage both following written instructions and VR training. User performances have been evaluated by time and errors count. User experience and system perception have been evaluated by a questionnaire and a NASA-TLX-like survey. VR training has not always demonstrated more effectiveness but all the users gave the VR training a higher score, meaning a high grade of acceptance of VR technology. The proposed method is interesting, especially for the knowledge extraction and know - how transfer to new users, but the proposed method is limited by the simple geometry of the components to be assembled, feature that is not realistic. Moreover, the authors do not describe the assembly models extracted from the data mining process, hence it is not clear how the knowledge extraction algorithm works. In [34] the authors describe a VR training system for medical devices. The training system has an intelligent agent that supervise and correct the user during the assembly sequence. The paper compares this system with other VR training system, like desktop and tablet based, without intelligent supervisor and with traditional training. The parameters used to evaluate the performance are completion time and user score. VR training without intelligent supervisor and traditional training has no significant differences, with only two minutes of difference of completion time. The VR system with intelligent supervisor outperforms the other two methods with a total reduction of 30% in the entire process. Moreover, the authors compared the use of VR HMD system with different input methods (hand motion tracking glove, Leap Motion controller, joystick) with the desktop and tablet based VR systems. The results show that the solution HMD with Leap Motion Controller obtained the best score in all the scoring categories: level of immersiveness, control of VR objects, realism, comfort, enjoyment and potential for VR training development. The solution HMD with hand motion tracking glove obtained the same results except for comfort level. In [35] the authors propose a VR training system to dismantle a research nuclear reactor. Thank to VR simulation, it has been possible to identify design problems that would have led to costs and time increment. Several games have been created to develop specific skills required during the dismantling (spatial awareness, depth perception, object manipulation). During the games, time and accuracy are evaluated in order to assess the user performances. After the skill development games, the users can use the simulator to disassembly the reactor core, in the order the user deems to be appropriate. 25 participants were evaluated and were asked to answer a custom evaluation questionnaire regarding the VR system. The authors distinguish the answers of expert user from the ones of not expert user. It is worth to note that expert users have given a lower rating regarding the comparison with the real – life master – slave manipulator system. This indicates that VR training systems are a useful tool for novice or inexperienced users and should be used as complementary tool for traditional training. The work in [36] describes the implementation of AR for user manual. The proposed solution aims to help the user performing the movements required to complete the assembly. The paper aims to develop the assembly skills of the user, without a knowledge of the functions of the parts to be assembled. This can be useful for novice and inexperienced workers, who do not know the product, to raise their efficiency and diminish assembly time. The work in [37] describes one of the few research on skill development in Architecture, Engineering and Construction through immersive technologies. The paper studies the effectiveness of AR in the construction of electrical conduits. The participants were chosen among a construction company and pre – test and post – test were carried out. The AR performance of each user is recorded and analyzed by a behavioral analysis software. The participants were asked to build an electrical conduit with HMD and with traditional paper plans. The user experience evaluation is assessed with pre and post questionnaire. The behavioral video analysis consists of a time comparison between AR experiment and the paper plan experiment. Three parameters based on completion time were identified: time of understanding, time of assembly, time of placement of the conduit. All the time parameters were less for the AR experiment. Errors were identified as performance evaluation parameter. Correct final layouts assembled with AR were 30% more than correct final layouts assembled with paper plans. A possible limitation of this paper is that the experiment has been performed in a controlled environment, not an actual building site. For this reason, problems like safety and work congestions are not take in account for the performance evaluation. However, the paper is not a completely laboratory experiment since the users are selected among real workers. Moreover, the conduits to assemble are taken from real building cases. Lastly, in [38] a comparison between text manual, training videos and VR screen based training is carried out. The authors used galvanic skin response and Average Heart Rate (AHR) to assess mental workload of the worker during the training and the performing of assembly tasks. In particular, participants trained with text manuals and videos made more errors than VR trained participants, regardless of the complexity of the task. Training times with text instructions and manuals were longer than the time with VR training. The authors suggested that VR training makes easier to comprehend abstract concepts and make the understanding faster by making those concepts more tangible. Text manual training showed the highest level of stress and mental workload both for complex and simple assembly tasks. Traditional video training and VR training did not show differences in stress level for simple tasks but VR show slightly lower mental workload than video. The authors suggested that this can be explained with the assumption that simple tasks do not require complex training methods, hence a video training has the same results of VR training. However, VR training performs sensibly better than other methods for complex tasks.

Chapter 3

Materials and Methods

In order to introduce high-level rules in an immersive direct modeling environment of assembly features and assembly constraints, it has been necessary to choose a game engine and 3D modeler. Unity3D was chosen as a game engine as it has become the standard for immersive industrial applications. Regarding the modeling of constraints and assembly rules, Grasshopper was chosen as a high-level rules environment. Grasshopper is a plug-in of Rhinoceros software and has a very active community of developers. To allow communication between Unity and Grasshopper, built-in components of the two software were used in order to set – up a User Datagram Protocol (UDP) server - client. At the same time new custom scripts have been developed to allow the user to interact with the model in the immersive environment and to send the suitable data to the modeling environment.

3.1 Unity3D Game Engine

Unity3D is a powerful 3D engine that allow the development of AR and VR applications. Since Unity is cross – platform, the developer can develop the application for several device and hardware set up, from desktop set up to mobile and immersive set up. Unity3D comprises a graphic engine, a physics engine and a live game preview. The graphic engine allows to complex physically based shaders (PBS) and a large range of shaders can be found online. It is worth noting that users can create their own shaders thanks to the shader scritps. The physics engine allows to simulate physical laws and physical behaviors. The live game preview enables to visualize the development and edits of the application in real – time while coding. Nowadays, Unity3D is one of the most used game engines both in terms of game development and research. With the arrival on the market of hardware for immersive applications, Unity3D has seen a remarkable development. The reasons for this success are many. First of all, Unity has a lot of resources to learn from, the learning curve is quite fast thanks to the use of the C# high level programming language. Unity3D is a great tool for rapid prototyping of applications, which is extremely useful in the research field. Unity is written in C language but user can develop applications using C# or Javascript, which are easy to learn. Since the HoloLens release, Unity has been chosen by Microsoft has the platform to use to develop Mixed Reality applications, and many toolkit has been released by Microsoft to allow user getting started with the development, such as Microsoft HoloToolkit and the most recent Mixed Reality Toolkit. Thanks to this tools, researcher can easily develop prototypes of immersive applications

3.2 Head Mounted Displays

Hardware for AR applications should be selected depending on the purpose of the final application. Head Mounted Displays (HMDs) can be Optical-See-Through (OST) or Video-See-Through (VST). The OST technology is based on a beam splitter capable of merging the virtual object with the real-world onto a semi-transparent lens. While, in VST additional cameras acquire the external scene which is then fused with the synthetic image in the internal screens of the VST device. Using OST, the user views the real-world without any intermediate means; when VST technology is considered, the user watches the real-world through the display [39]. A second difference among AR hardware features can be found in the processing unit. Some devices can be used standalone because they integrate the necessary electronics to collect data from sensors and visualize the results; others need additional hardware connected to them by cables to elaborate information. About VST is worth taking about HTC Vive Pro. The HTC Vive Pro is not standalone, but is one of the most powerful in terms of display specifications. The HTC Vive offers 1080x1200 pixel resolution for each eye, 2160x1200 in total. It has a 90Hz refresh rate, thus ensuring the frame rate is high enough to prevent motion sickness and provide a smooth experience overall. It offers a 110 – degrees FoV. HTC Vive supports stereographic 3D view and such a large FoV enables the user to look all around. In addition to that, HTC Vive Pro is equipped with tracked controllers that can be used inside the VR environment to interact with objects. VTS are usually chosen for Virtual Reality applications, but this don't prevent their application to Augmented Reality purposes: for example, HTC Vive Pro can stream the images captured by the external camera to the screens. On the other hand, OST are based on a different technology that fits perfectly for AR. In the past few years, different projection techniques have been proposed for OST HMDs [40]. The half mirror-based HMD can be considered a simple solution, but this technology offers a limited Field of View (FoV). Another solution, which can be considered more sophisticated, is represented by the use of convex or free-form mirrors that are capable of widening the FoV. This solution is implemented in the Meta2 device: Meta Company gave straightforward importance to obtaining a larger FoV and better display resolution. The waveguide technology is the last technology available on the market. This technology is based on the laws of optical reflection and optical refraction. The diffraction grating, likewise a lens, bends the light into a thin sheet of glass or plastic. The light bounces through the pipe until it exits towards the eye. This solution is implemented in Microsoft HoloLens. Over the last few years, many companies have introduced their own AR device: the first were Google Glass by Google; over the years Vuzix has produced several types of eyewear devices, some conceived as peripherals of a host device, others to be used in independent mode; up to the latest Microsoft HoloLens. Microsoft HoloLens and some models of Vuzix Glasses embed computational power. Microsoft HoloLens are equipped with an Intel Atom Processor as well as Vuzix M300 Smart Glasses, for instance. Microsoft HoloLens is a completely self-contained computer that the user wears on his head. Besides, Microsoft HoloLens uses holographic lenses, called waveguides, while Meta2 uses a convex mirror for the projection of images in front of the user's view. Thanks to this original hardware, Meta2 offers a 90 degree FoV ensuring a more immersive experience compared to HoloLens. The HoloLens as well ensures a stereographic 3D view, but it forces the user to keep his eyes facing strictly forward since the FoV is only 30-degrees. When a comparison between HoloLens and Meta2 is addressed, it is worth noting that the type of gestures implemented in

Meta2 device are more natural than those available in HoloLens. In fact, it is opinion of authors that using the whole hand to grab a hologram, or fingertip to select icons in the screens, is easier than the "blooming" gesture, or the air tapping fingers movement implemented in HoloLens. On the other hand, the hand tracking and gestures recognition capabilities of Meta2 are not so effective and easier gestures, even if less natural and intuitive like the HoloLens gestures, ensure a more reliable interaction and a more satisfying User eXperience (UX). The tracking system represents another significant difference between VR and AR devices. Vive is equipped with a 6 DOF IR Laser-based 360-degree tracking system with two external sensors with 4.5x4.5 meters tracking area. An entire room set up for a Vive is recommended in order to improve UX, but HTC Vive can be set – up also in the simple standing mode, whereas the HoloLens tracking system is completely enclosed. Microsoft HoloLens tracking system is based on a different technology since it is fully embedded into the device. It relies on four "environment understanding cameras," two on each side and a central depth camera. The "environment understanding cameras" provide the basis for head tracking, and the Time of Flight (ToF) depth camera aims at two objectives: it helps the hand tracking and performs surface reconstruction, which enables the positioning of holograms on physical objects. It's fair to say that Microsoft reached quite high standards in terms of tracking. On the other side, the Meta2's tracking system gives a 6DOF positional tracking: it is similar to that used by HoloLens because it is based on a depth camera, webcam and IMU. The integration of data from the depth camera and the webcam with data from the IMU is obtained through a proprietary algorithm that performs a sort of Simultaneous Localization and Mapping (SLAM) [41]. In parallel, and sometimes even earlier respect to the development of wearable devices, companies developed external devices to bring the real and the virtual world closer: for example, Microsoft Kinect and LeapMotion controller belong to this set of devices. Focusing the attention on Microsoft Kinect, it uses the same technology implemented in Microsoft HoloLens: in fact, it is equipped with a webcam coupled with an IR sensor that allows the reconstruction of the surrounding environment. LeapMotion controller is a device designed specifically to allow interacting with virtual objects, without the need for dedicated gestures or controllers, but requiring an intuitive use of the user's hands. The LeapMotion controller efficiently merges user's hands into a virtual environment. The device guarantees a very accurate tracking as it blends data from two cameras and three IR sensors. A hemispherical space above Leap Motion of approximately 40 cm in diameter is the best tracking area. In some applications, these devices have been coupled with wearable equipment to create more complex interactive simulations [42]. The evolution of the VR and AR dedicated hardware aims at reducing the gap between the virtual world and the real world, and at implementing the concept of digital twin where there is a virtual copy of a real object and the user can switch between them in real time.

3.3 Grasshopper

Grasshopper is a plug – in of Rhinoceros 3D modeling software. More specifically, Grasshopper is a high level visual programming environment. Being a visual programming language means that all operations on geometries are operated by graphic elements, i.e. blocks, within the grasshopper Graphical User Interface (GUI). Every block represents a node of the grasshopper graph and every block works as a function of a computer programming language. Every node performs specific operations, given the correct set of inputs, and generates one or more outputs. Inputs are on the left side of the block, while outputs are on the right side of the block. The data supported by Grasshopper are several. Data can be geometrical data or non – geometrical data. Geometrical data are: points defined by a (x, y, z) parameters; vectors defined by a direction and a magnitude starting from a point; lines i.e. a segment between two points; NURBS curves and NURBS surfaces, B – Rep geometries and meshes. Non – geometrical can be: text string; integers and floats or Boolean variables. Besides the nodes that work as function, Grasshopper has nodes that work as input variables. Input nodes can be container nodes of geometrical data or parameter nodes. Container nodes do not perform any operations or do not operate any function, they simply import geometry from the Rhinoceros viewport into the Grasshopper graph. Parameters nodes are used to set non – geometrical data as input of subsequent Grasshopper blocks, like slider to set a numerical value. Each node is connected to others by mean of wires and the output of a block immediately become an input for the following block. In this way, the data flow is unidirectional and as soon as each block receives the input, the function of the block is operated and the geometry is created. This dataflow is similar to procedural programming with the difference that in procedural programming the operations are executed linearly, while in Grasshopper visual language the operation is executed as soon as the input is received by the block. The blocks that operate functions can be divided into:

1. Blocks that generate new geometry, points, vectors, B – Rep, etc.;
2. Blocks that transform already modeled geometry, for example rigidly moving a cube in the space;
3. Blocks of calculations based on geometry or numerical input, for example calculating the volume of a closed geometry.

In Grasshopper the data between each block can be either a single data, like a numerical value, or set of data. This can be useful when working with series of data describing some recursive feature of the geometry, or when working with sequence of the same data changing over time. In this case the wires connecting the blocks is not a single wire, but there is a double wire connection meaning that the data between the two blocks are multiple. This specific feature is useful in our research since the communication between Unity3D and Grasshopper involves the streaming of sequential data via UDP. Among the large number of open source Grasshopper components that are available on internet, we selected and used two particular components: gHowl and Kangaroo. gHowl component is a component that allows to receive User Datagram Protocol text data from an external sender. Kangaroo is a physic engine component for simulation inside Grasshopper. These two components are the component on which the high – level modeling rules system between Unity3D and Grasshopper is based.

Chapter 4

Immersive Modeling Environment for Assembly and Disassembly Design

In this chapter we describe the development of a modeling environment within Unity3D, thanks to the implementation of several commercial tools, ProBuilder and Grasshopper. The modeling environment is connected to Rhinoceros Grasshopper in order to enlarge the modeling capabilities of the ProBuilder built – in classes and methods.

4.1 Immersive modeling environment

The Unity ProBuilder plug-in was used to create and edit the geometry within Unity3D. We mainly focused on creating and editing geometry necessary for coupling the components, for example modeling pins or moving coupling surfaces on the imported model. ProBuilder allow the user to create and edit custom geometry directly in the Unity Editor. ProBuilder is particularly suitable for the aim of this research because it allows to create and edit meshes and also because ProBuilder has a comprehensive Scripting Application Programming Interface (API). Hence, the user can write its own C# scripts to develop specific tools and functionalities. Another advantage of the use of ProBuilder is that it comes with a large library of meshes of primitive geometries, like cubes, cylinders and planes, which is very useful since the mating features between components are essentially primitive geometries, like pin/hole connections. Moreover, it has Boolean Constructive Solid Geometry (CSG) tool which implements intersection, subtraction and union of two meshes. Whit this last function, the user can create on-the-fly custom geometry for a coupling element on one mesh, merge this new geometry to the component mesh and then send all the data to Grasshopper via UDP.

4.1.1 Interaction in ProBuilder

ProBuilder meshes works like any other Unity GameObject. This means that the user can apply rigid movements, rotations and scale factors to the entire mesh. In the Unity Inspector the user can add components to those meshes and attach scripts in order to control or add properties to the meshes. Furthermore, different selection modes are implemented. Object mode represents the standard Unity selection mode for selecting and editing GameObjects as a whole. Vertex mode, Edge mode and Face mode are the selection modes for selecting and editing vertices, edges and faces on a ProBuilder mesh. In such a way the user can select specific entities of the mesh and precisely select edges and faces to mate the two meshes. It is worth noting that all the ProBuilder commands work only with ProBuilder meshes. Imported meshes or 3D Unity objects do not support ProBuilder functionalities. However, in the scripting API it has been implemented a conversion tool to convert imported or Unity built – in meshes into editable ProBuilder meshes.

4.1.2 Mesh creation in ProBuilder

ProBuilder offers three ways to create new geometries. Firstly, the user can instantiate primitive geometries from a built – in library. Pre – defined shapes comprehend cubes, cylinders, but also more complex geometries like toruses and icospheres. Each pre – defined shape is fully customizable in the geometric properties, for example the radius of cylinders and icospheres, and in the Level of Detail (LOD), for example user can set the level of subdivision of icospheres, in order to meet hardware specs and not to slow down the application. In Fig. [Fig. 4.1] it is depicted an assembly of a piston rod. The two pins, coupling the piston rod with the piston cap, have been modeled by instancing two cylinders and placing them in the holes of the two components. The great advantage of this geometry creation method is that the user can insert parameters, such the cylinder radius for example, to create geometries with the exact dimensions he desires.

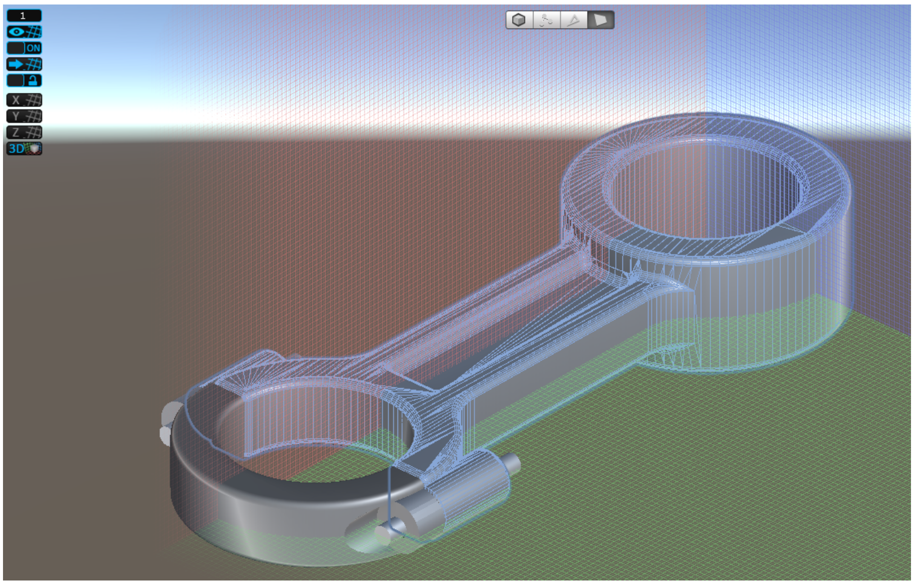


FIGURE 4.1: The assembly of a piston rod with the two connecting pins modeled with ProBuilder suite

The second type of mesh creation is the polygon shape creation. The workflow of mesh creation with the polygon shape tool is similar to the sketch and extrude workflow of CAD software. The user creates custom 2 – dimensional shape, on a plane or on a face of a pre – existing mesh, and then extrude the 2D sketch to create a 3D mesh object. The polygon shape tool is interactive, while instancing a primitive geometry represents a more static interaction. The main advantages of polygon shape tool respect to the instancing of primitive geometries from the built – in library are: firstly, polygon shape is interactive and the user can create new geometry following the shape of mesh; secondly, geometries modeled with the polygon shape tool are indefinitely editable because the user can edit the control points of the 2D initial sketch. For example, in [Fig. 4.2], the white shaft has been modeled following the perimeter of the piston hub and then extruded with a simple drag movement of the input hardware. The drawback of the interactivity of polygon shape tool is that the user cannot set the exact dimensions of the geometry. However, this is a minor drawback since in the concept phase, exact final dimension might not be already known by the designer.

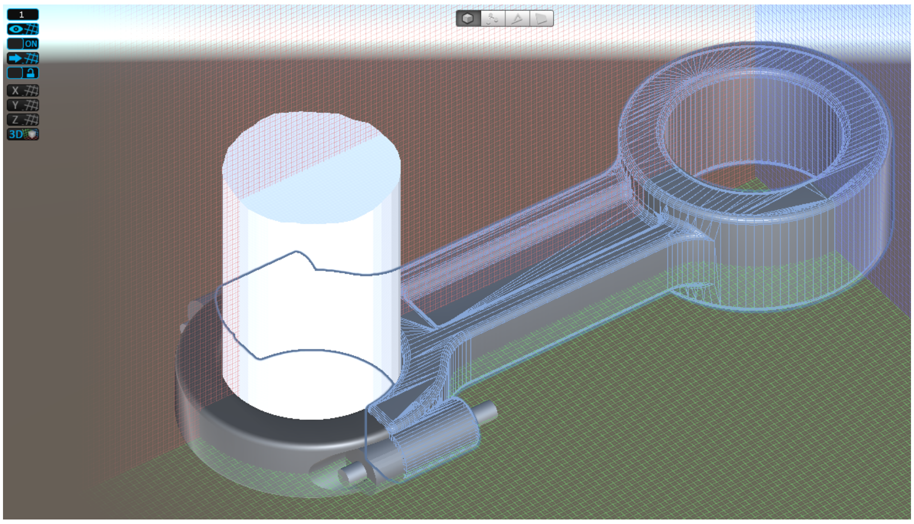


FIGURE 4.2: Modeling of the shaft following the perimeter of the hub

In order to place new geometries in the world coordinate system we used the Unity plug – in ProGrids. With ProGrids the user can place 3D objects, imported or modeled on – the – fly within Unity, with precision: the object snaps to the world grid. The position of world grid always remains the same. Hence, the user knows where components are placed, the distance between them and the magnitude of a movement. With these capabilities, ProGrids allows the user to keep objects aligned, for example with symmetry planes [Fig. 4.3], or at a certain distance.

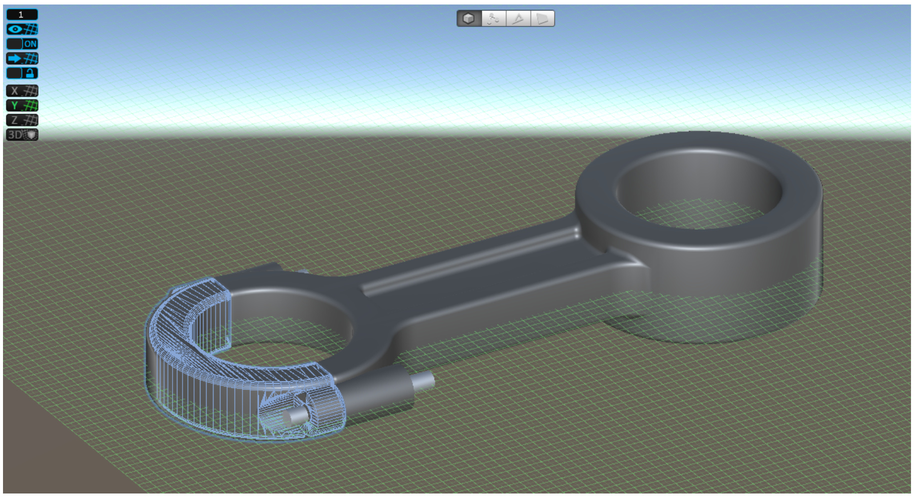


FIGURE 4.3: Symmetry plane of the piston rod highlighted by ProGrids

The snapping value depends on the grid units and it can be set by the user, accordingly to his design intent. Since the application is conceived for mechanical components, the default grid unit are millimeters and the default snap value is 1 mm. However, for more precise edits or modifications the snapping value can be reduced.

4.1.3 ProBuilder scripting API

ProBuilder provides developers a scripting API in order to enlarge ProBuilder tools and functionalities by writing custom C# scripts. The API includes three namespaces. The namespace UnityEditor.ProBuilder provides classes for Unity editor integration. This namespace and its classes can be used by the developer to create custom User Interfaces (UI) with new menus and toolbars for specific commands. The namespace UnityEngine.ProBuilder provides classes for low level mesh compiling. These classes allow the user to create meshes, handle events on vertices, edges or faces of a datum mesh and operate mathematical operations with meshes. The namespace UnityEngine.ProBuilder.MeshOperations provides classes for mesh editing. Through these classes the user can edit meshes and manage input/output events. It is worth noting that libraries from UnityEngine.ProBuilder and UnityEngine.ProBuilder.MeshOperations, which are the libraries required for creating and editing meshes, are available at the RunTime. Hence, all the scripts implementing classes from those libraries can be activated by input and executed in the PlayMode, or RunTime.

4.1.4 3D modeling in RunTime

In order to create new geometries in the RunTime, we developed a simple UI for the touchpad of the Vive controller. The UI is a radial menu and the user can access either the polygon shape tool or the primitive shapes library by using touch input with his finger. To help visualize the position of the finger in the VR scene, we highlighted the finger position with a white sphere over the virtual touchpad. Once the creation mode of new geometry is selected, a red raycast helps the user to aim where pose the control points of the 2D sketch. Control points are added sequentially by hitting the trigger button [Fig. 4.4].

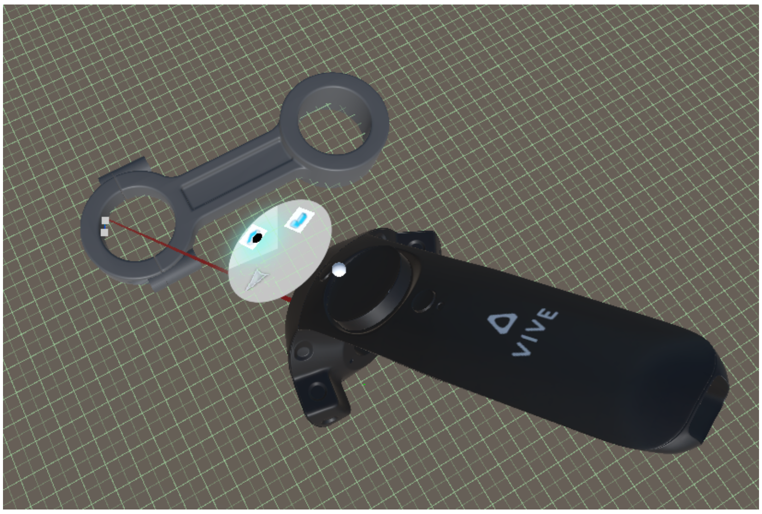


FIGURE 4.4: Sketching with ProBuilder with HTC Vive controller

Once the 2D section is closed, by hitting the first control point with the raycast, the user can extrude the new geometry to the desired height by dragging the control point in the center of the 2D polygon [Fig. 4.5].

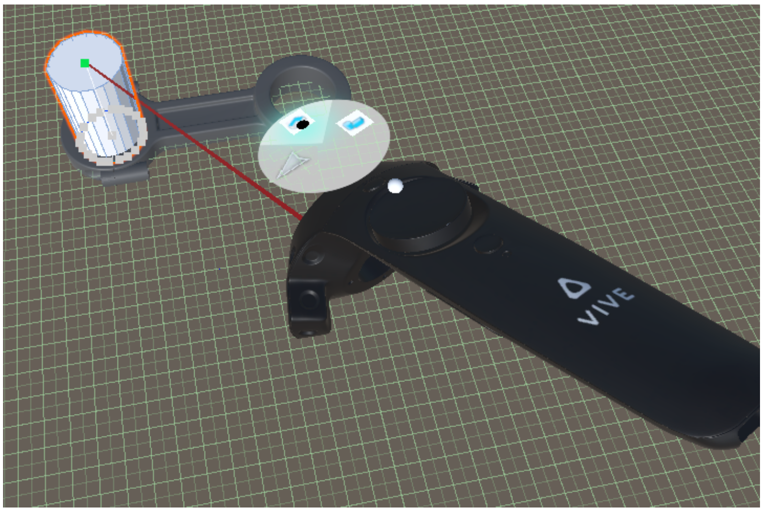


FIGURE 4.5: Extruding the 2D sketch with ProBuilder with HTC Vive controller

Similarly, the user can model new geometries chosen from the library of primitive forms [Fig. 4.6].

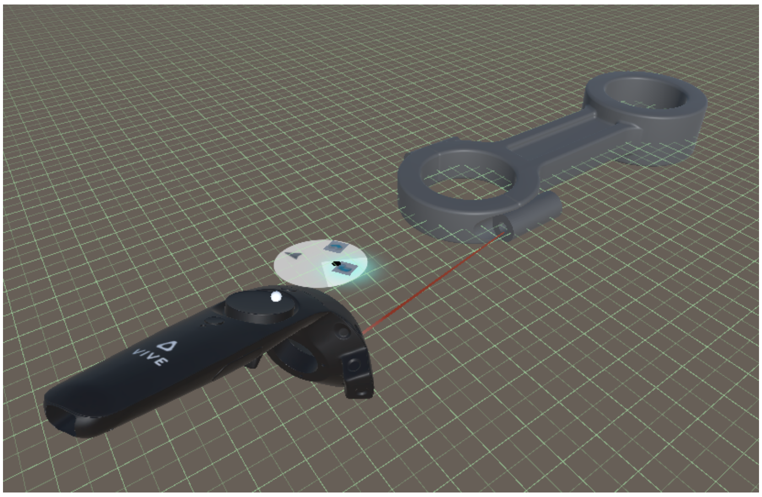


FIGURE 4.6: Rough positioning of the primitive

With this mode, you can create primitive forms with exact parameters, for example by entering the value of the radius of a cylinder or the length of the side of a cube [Fig. 4.7].

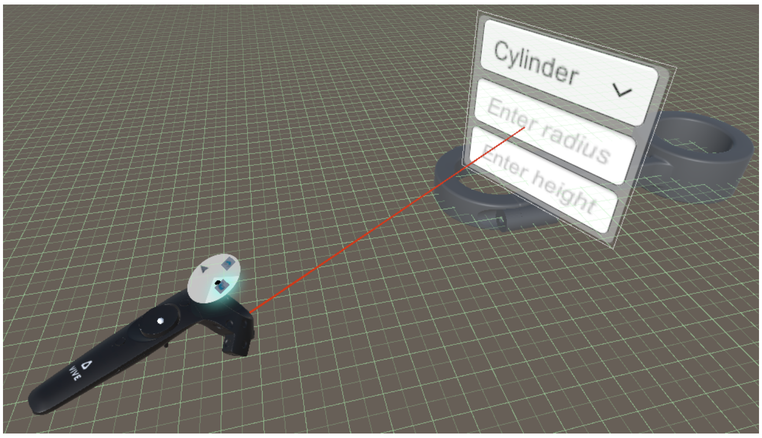


FIGURE 4.7: Setting the parameters for the primitive definition

Once the new geometry has been instantiated, the user can drag and pose the solid in the final position [Fig. 4.8].

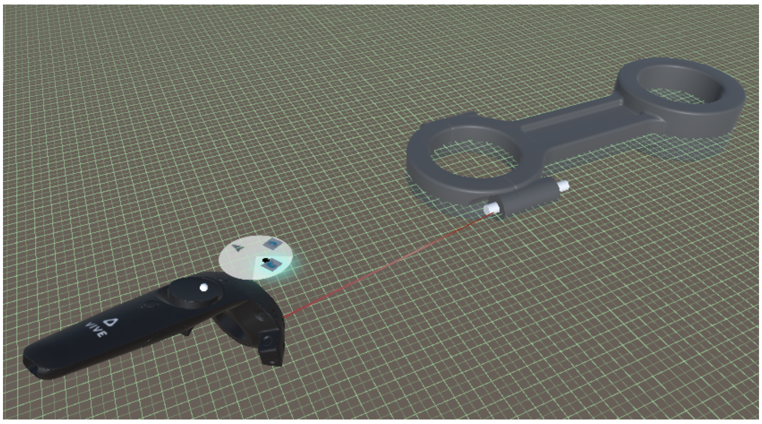


FIGURE 4.8: Final positioning of the primitive

4.1.5 Selecting geometry in RunTime

As for modeling new meshes, the user can edit and modify single control points, edges and faces of a mesh. In order to select single entities on a mesh, several methods from the class SelectionPicker have been used in the scripts attached to the UI. The class SelectionPicker offers two types of selection: direct selection of a single element, or selection of multiple elements within a selection rectangle. However, the direct selection is available only for single faces through the method PickFace, which returns the first hit face on a ProBuilder mesh, given the screen position of the camera. Edges and vertices are selectable only within a rectangle of selection, given the position of the camera. Since the common selection method of HTC Vive controllers and Microsoft HoloLens is based on raycastin, i.e. projecting a ray from the tip of the controller or from the gaze, we developed a system to bypass the limitation of the rectangular selection. When the ray from the generic input device hits the object, the mathematical description of a rectangle is created at the end of the ray, centered in the hit position. This fake rectangle is instantiated as a single event GameObject. This rectangle is then used as input parameter in the methods PickEdgesInRect and

PickVerticesInRect. The limitation of this method appears in the selection of two too close elements. In order to select the right one, the user has to go closer, or scale the object, so that the two close elements can be selected without uncertainty.

4.1.6 Editing geometry in RunTime

Thanks to the SelectionPicker method the user can select entities to modify the geometry. Under the namespace UnityEngine.ProBuilder.MeshOperations are listed all the possible classes to edit geometry. With the class SelectMode it is possible to switch between the selection modes of ProBuilder: vertices, edges, faces or the whole object. Depending on the active selection mode, the user can edit the geometry by manipulating single vertices or entire faces. For example, when in vertex selection mode, he can edit the initial 2D section of an extrusion by dragging each vertex, in order to modify the entire profile of the extrusion, or he can manipulate a vertex on the side of the extruded mesh [Fig. 4.9].

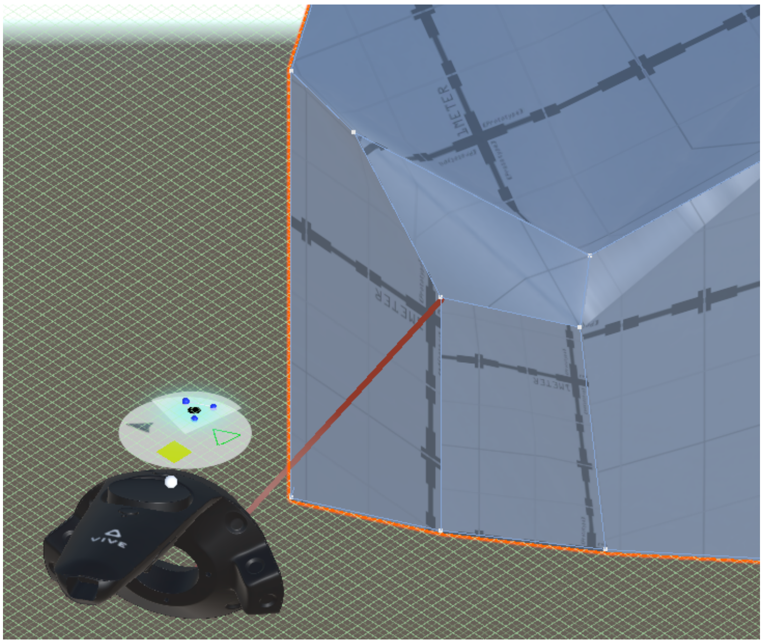


FIGURE 4.9: Single vertex editing by dragging command

In the same way, he can edit the entire face of the extrusion when in face selection mode [Fig. 4.10]. The user can perform operations push / pull on vertices, edges and faces to modify their positions and orientations; while scaling operations are available only on edges and faces.

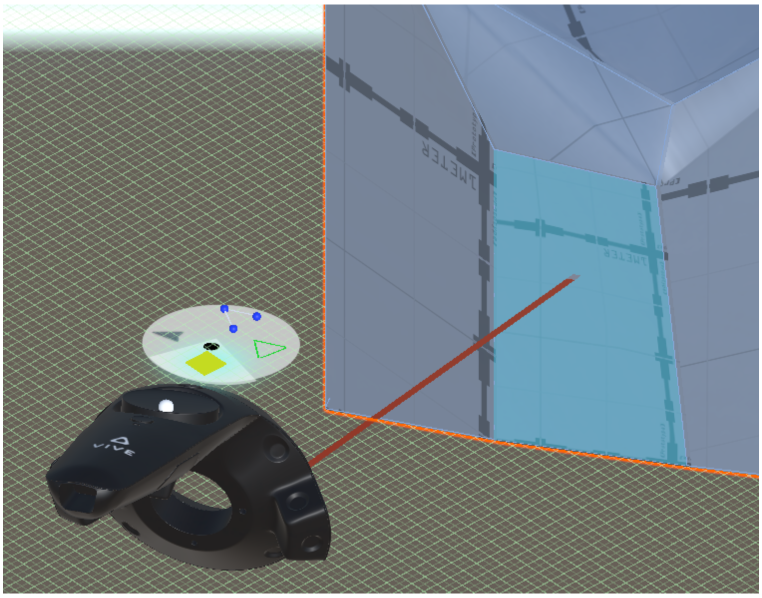


FIGURE 4.10: Face editing by dragging command

The class AppendElements allow the user to append elements to meshes. The method AppenVerticesToEdge enables the user to append new vertices to edges. This method can be useful to subdivide an edge and to add finer details. For deleting a face, it has been used the class DeleteElements class, and then the method DeleteFace. For example, the user can firstly subdivide the mesh by inserting new vertices. Then, he can delete one of the new faces and then bridges the two edges to create a through-hole [Fig. 4.11].

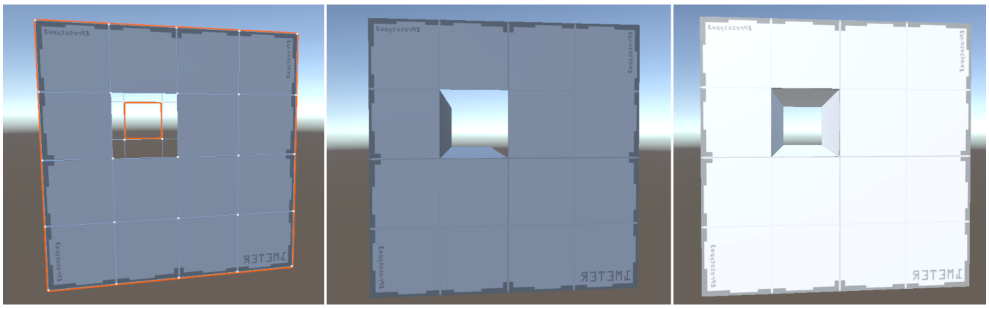


FIGURE 4.11: Left: deletion of two faces of the polygon. Center: bridging of the edges of the hole. Right: final hole on the face

It is worth noting that the built – in algorithm of edge bridging recognizes and poses the new faces with the correct normal orientation [Fig. 4.11, Right]. Another possible operation very frequent in the mechanical design is the chamfering. ProBuilder offers a class for chamfers creation: the class Bevel. The method contained in this class that allow to create chamfer is BevelEdges. This method can be applied equally to a single edge or a group of edges, like in [Fig. 4.12, Left], if the user is in edge selection mode; or it can be applied to a whole face, like in [Fig. 4.12, Right], if the user is in the face selection mode.

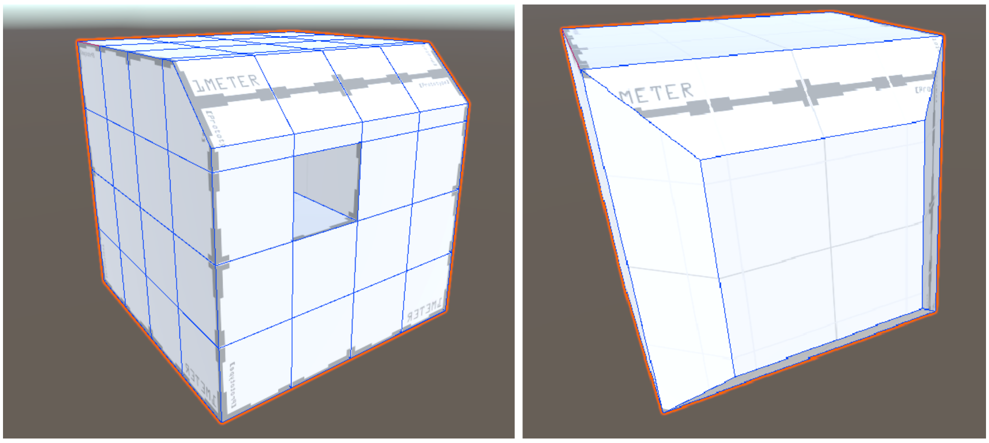


FIGURE 4.12: Left: single edge chamfering. Right: all edges chamfering

4.2 Communication Module

Unity represents a great tool for developing immersive applications. With the use of plug – in like ProBuilder, it is also possible to model and create new geometries in the runtime. Even from the physics simulation perspective, it is a powerful tool since it has two built – in physics engines: one for 3D simulation, one for 2D simulation. Unity physics engines provides developers with components capable of simulating physical behavior. However, these geometrical modeling and simulation capabilities are not enough from an engineering point of view. It is necessary to introduce in Unity more precise and accurate geometrical modeling rules and physics behaviors. For this reason, the Rhinoceros' plug – in Grasshopper has greater modeling capabilities and a great amount of components for physical simulations. Hence, we developed a communication module to send data, geometries and input from Unity into Grasshopper's modeling language. All the Grasshopper's calculations run in the background on the Unity application. Hence, a stable and fast system for data transfer between Unity and Grasshopper is fundamental. We opted for a method of sending and receiving data from Unity and Grasshopper based on User Datagram Protocol (UDP). The major drawback of the UDP compared to Transmission Control Protocol (TCP) are that there is no sequencing of data in UDP and that delivery of data to the destination application cannot be guaranteed. On the other hand, UDP is lightweight, it is faster than TCP and it is simpler to set up even by beginner users.

4.2.1 Data exchange workflow

In order to enable the data exchange between Unity and Grasshopper, a UDP server – client network has been set up. On the Unity side, we used the namespace System.Net in a C# script to send and receive data. This namespace allows to set the IP address and the port. On the Grasshopper side, we used an open source add-on component of Grasshopper called gHowl which allows to receive and send UDP messages. The data workflow consists of the following steps. First, the user generates geometry in Unity through the ProBuilder tools, either through instancing primitive shapes or through the polygon shape tool. Then the key values of the new geometry are collected and encoded in a single text string. Each input value is separated using a semi-period from the following value. The text string is sent via UDP client to Grasshopper. The transferred string is composed by three subset of data and the pattern of the string varies depending on the type of geometry creation tool used by the user. In Grasshopper the text string is parsed, using the split component to separate each value from the others. Then each subset of data is listed in order to become the input parameters for the geometry components in Grasshopper. The geometry is then modeled in Grasshopper. Lastly, the geometry can be modified through the Grasshopper components or through scripts and then is sent back to Unity via UDP as well.

4.2.2 Data exchange of primitive shapes

In the following steps, we describe the workflow for the creation of a cube through the primitive shape library of ProBuilder and the exchange data between Unity and Grasshopper. On the input event with the input device, for example the HTC Vive controller, the string of data is sent to the UDP receiver in Grasshopper, which has been set up with the IP address and the port to listening to [Fig. 4.13]. The output of this block is the parameter D and it consists of a text string, since the Unity UDP sender is sending a text string.

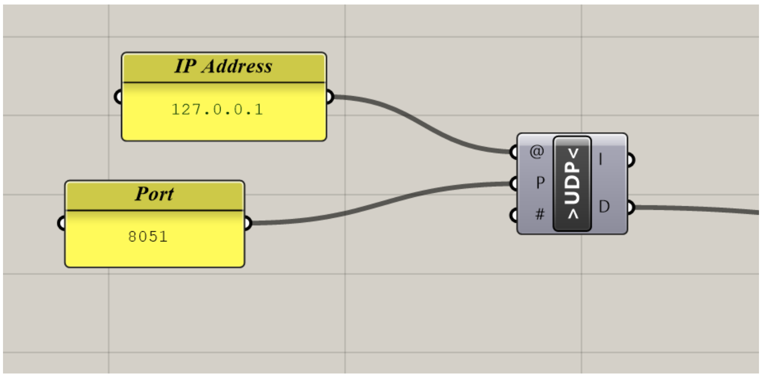


FIGURE 4.13: UDP receiver block

The text string is composed by three subset of data. First parameter is the type of geometry, listed accordingly to the dropdown menu of primitive shapes in ProBuilder: 0 for cubes; 3 for cylinders; 4 for planes; 7 for cones; 10 for icosahedrons. In this example the first parameter is 0 because the user is modeling a cube. The text string is then unpacked through the split component. The split component accepts a text T and a separation character C as inputs. The split component divides the text string accordingly to the separation character. Unity uses semi-colon as separation character. The output of the split component is the resulting text fragments R and these fragments are lists [Fig. 4.14]. The fragments are listed and through the Sort Item component it is possible to retrieve a specific item from the list. In the Sort Item component with index = 0 it is retrieved the type of geometry. In the Sort Item components with index = 1, the dimensions of the cube are retrieved. In the Sort Item component with index = 2 the position of the geometry is retrieved, accordingly to the position of the ray generated by the raycast.hit method of HTC Vive controller.

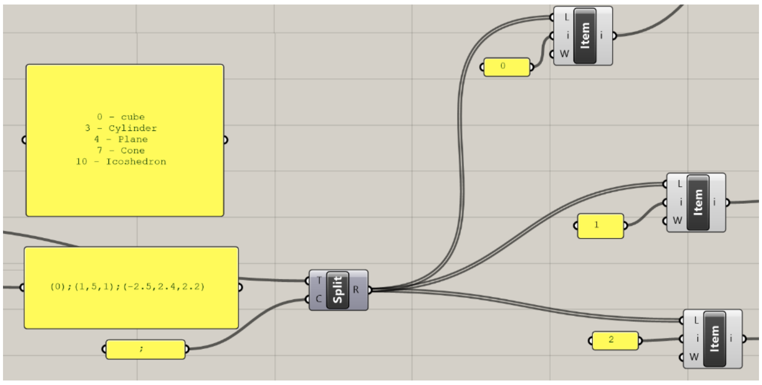


FIGURE 4.14: Message parsing with the split block

The list with index = 1 is then manipulated in order to convert text data, containing the dimensions of the cube, into a vector, containing the three dimensional parameters of the cube. In this example the cube is 1x1 mm with a height of 5mm. In the same way, the list with index = 2 is manipulated in order to set the plane and the center position of the cube [Fig. 4.15].

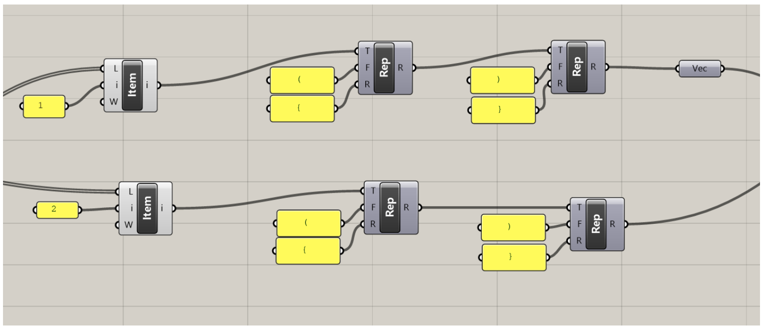


FIGURE 4.15: Conversion of the txt string into vectors

The plane where to pose the new geometry is selected by the user in the ProGrid set – up panel. In Unity, the plane XZ is the default horizontal plane and the Y axis is the vertical axis. Hence, by default the Grasshopper block chain implements the XZ plane as the sketching plane [Fig. 4.16]. However, the sketching plane can be switched upon user request. This can be observed in the Grasshopper diagram in the dashed lines in [Fig. 4.16].

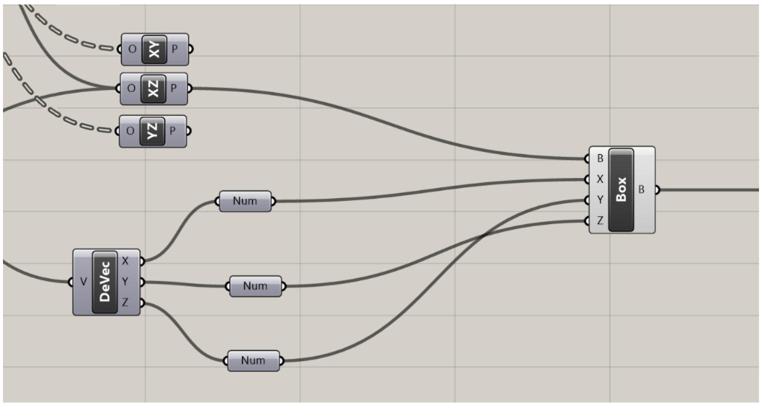


FIGURE 4.16: The box creation in Grasshopper

Once sketching plane is set and the text string is converted into vectors, the cube is created through the Box components which accepts the sketching plane as Base Plane B, and the three dimensional components as dimension of the box. It is worth noting that the Y component and Z component are switched because Unity uses Y as vertical axis, while Rhino uses Z as vertical axis. The result of the modeling workflow can be previewed in the Rhinoceros window. According to Unity standards, the x axis is the red segment; the Z axis is the blue segment and the Y axis is the green segment. In Rhinoceros a cube with (x,z) = (1,1) and height y = 5 is modeled and displayed [Fig. 4.17].

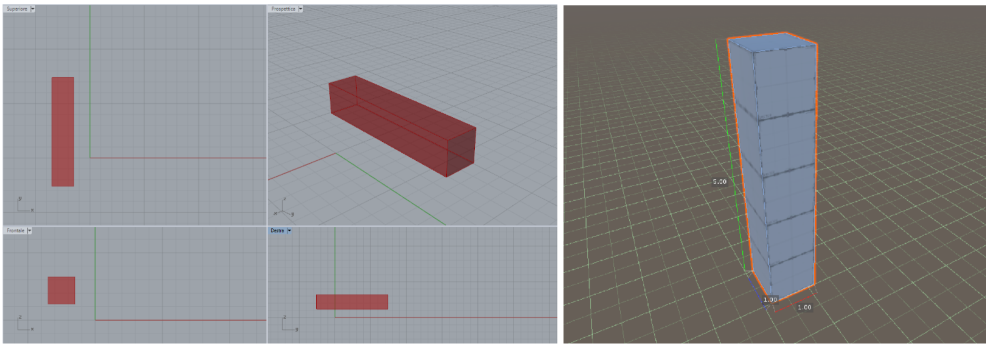
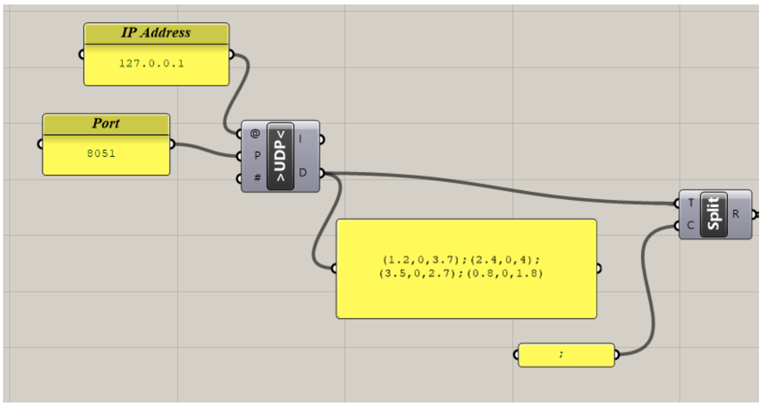


FIGURE 4.17: The modeled box in Grasshopper and in Unity3D

4.2.3 Data exchange of polygon shapes

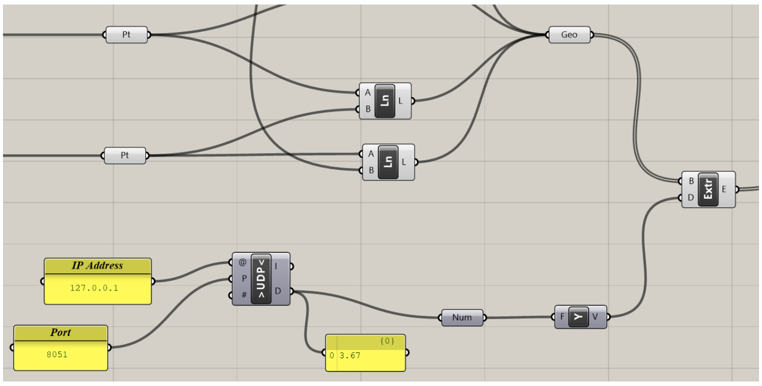
The Polygon shape tool data exchange is pretty similar to the primitive shape data exchange. The main difference consists in the form of the text string sent from Unity to the gHowl UDP receiver. The text string contains simply the coordinates of the control points of the 2D polygon used as profile for the extrusion [Fig. 4.18]. As you can notice, the selection of the Y plane as sketching plane is highlighted by the second component of the coordinates of the control points, which is always equal to zero.



```
(1.2,0,3.7);(2.4,0,4);
(3.5,0,2.7);(0.8,0,1.8)
```

FIGURE 4.18: The incoming txt from Unity3D containing the coordinates of the control points

The main difference between the two shape modeling methods consists in the second input that is transferred to set the height of extrusion. To extrude a 2D sketch with ProBuilder, the user has to select the control point in the center of the 2D polygon. The selection of the center control point represents a second event of the input device which is sent to Grasshopper via UDP. The dragging distance of the center control point represents the extrusion height for the polygon in Grasshopper. In this example the extrusion distance is 3.67mm as highlighted in the output panel in Grasshopper and in comparison figure [Fig. 4.19].



```
{0}
0 3.67
```

FIGURE 4.19: The UDP receiver receiving the extrusion height

The final result of the creation of new geometry with the polygon shape tool is represented in [Fig. 4.20]. According to Unity standard, XZ plane is represented by red and blue axis, while Y axis is in green. The same coordinates system is used in Rhinoceros because during the parsing of the data from unity, Y and Z axis are inverted. Hence, in Rhinoceros preview window XZ plane is the sketching plane and the direction of extrusion is along Y axis.

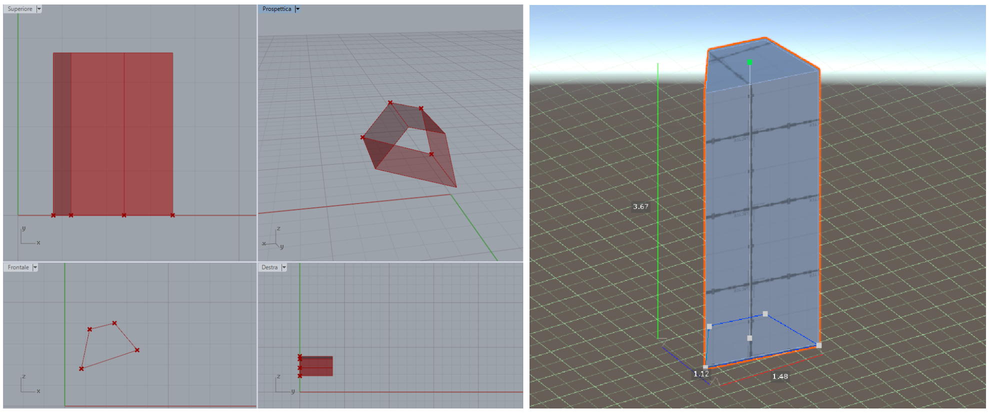


FIGURE 4.20: Extruding a polygon shape in Unity3D and the correspondent extrusion in Grasshopper

4.3 Immersive assembly constraints

After the design and the implementation of the Unity – Grasshopper integrated modeling suite, we started developing the modeling system of the assembly constraints. Rhinoceros does not support parametric modeling and for this reason Grasshopper has been integrated in Rhinoceros. Among the numerous Grasshopper plug – in, Kangaroo represents a physics and constraints solver for Grasshopper. Kangaroo offers to developers a set of components for form finding, simulation of physical phenomena and a solver for geometric constraints. The Kangaroo simulation engine is based on the Goal concept. A goal is a function that rules input parameters, i.e. points, edges and faces. Multiple goals can be applied to each object and the solver component has the capabilities to balance the goals that have to be match in order to meet all the constraints. The goals balance can be governed by the user by giving different weight to each goal.

4.3.1 Kangaroo Components

The components that allow the user to interact and set assembly constraints are: the solver component; the grab component and the rigid body component. The solver component is the component that collects all the goals and solves the system. In Kangaroo there are three types of solvers: simple solver, step solver and zombie solver. The simple solver runs and solves the goals at the user input through a generic Grasshopper Boolean toggle component. The step solver runs and solves the goals only when input is refreshed or at predefined time steps, for this reason the reset function is disabled. The zombie solver runs and solves the goals until the maximum number of iterations is reached. For our research, we implemented the simple solver component. The simple solver requires at least two input: the goal objects, i.e. the objects that will be moved accordingly to the constraints; and the reset toggle; mandatory to let the solver start the simulation. The drag component allows the user to move the simulation points in the Rhinoceros viewport. In our case, this component has been used to drag geometries in the coupling position, or to move the entire assembly accordingly to the assembly constraints. The rigid body component is used to identify B – Rep surfaces or meshes as rigid body. This means that the input surface acquires six degrees of freedom. Mechanical assembly mating constraints can be applied to rigid bodies by using the Kangaroo goals blocks. It is worth noting that if a B – Rep surface is used as input parameter of a rigid body block, an intermediate block of B – Rep deconstruction is required. The B – Rep deconstruction is necessary in order to be able to select the vertices, edges or faces of the B – Rep surface as input of the goals blocks [Fig. 4.21].

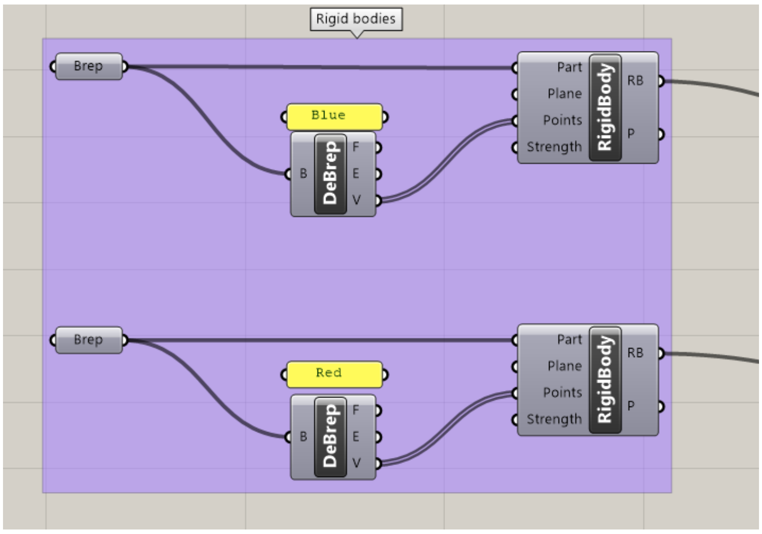


FIGURE 4.21: Decomposition of the geometry

4.3.2 Kangaroo goals library

In the following section, a concise overview of the most used kangaroo goals. We used these goals to define a series of block diagrams, one for each joint we want to simulate. Anchor goal: this goals keeps the entity position in its original position. As input the anchor block accepts the point P to lock and the target T to lock the point P to [Fig. 4.22].

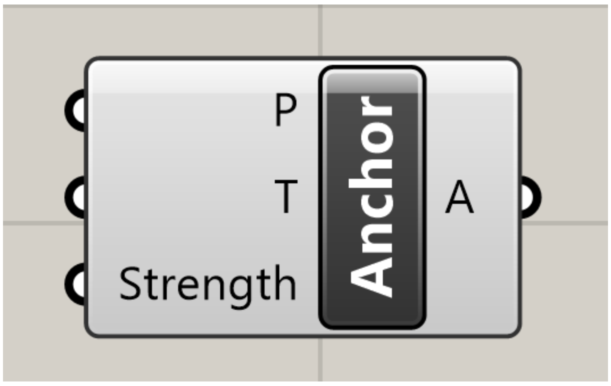


FIGURE 4.22: Anchor block to lock the position of the element attached

AnchorXYZ goal: similar to simple Anchor, this goal allows to decide for which of the world directions to restrain the point in. The AnchorXYZ block accepts a point P as input. To lock a direction, it is required a Boolean toggle to set True or False the corresponding input direction in the block. When the Toggle is set to true, the corresponding movement direction is locked. For example, to allow the movement of a point only along the vertical Y axis, the user would set X = true, Y= false, Z = true. This block is used to set the frame of the assembly, by imposing all the X, Y, Z variables true, as in [Fig. 4.23].

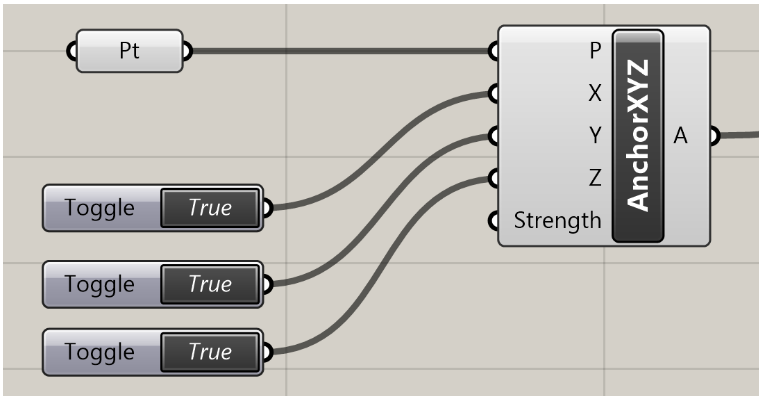


FIGURE 4.23: AnchorXYZ block used to frame the geometry

On Plane goal: this block is similar to Anchor and AnchorXYZ block since it locks input points on a plane. The On Plane goal is used to set the frame of the assembly. This block accepts point to lock as input, and the plane to lock the points on [Fig. 4.24].

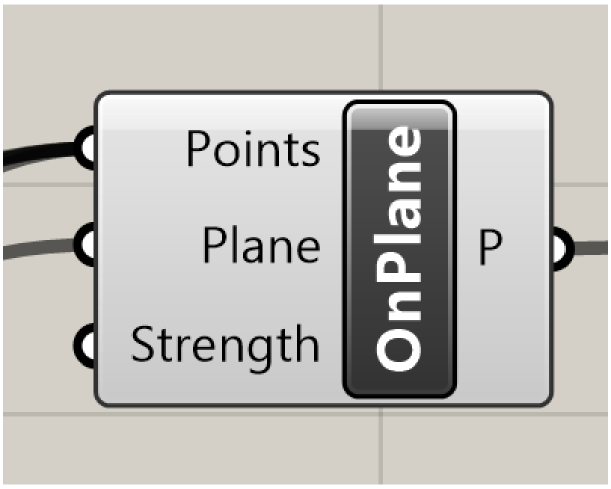


FIGURE 4.24: OnPlane block

Length goal: this block keeps two points at a predefined distance. This block accepts the conjunction line between two points and the length of the line as input. The length can be set by external parameters, for example by numerical 1D sliders, or by the positions of the controller sent via UDP [Fig. 4.25].

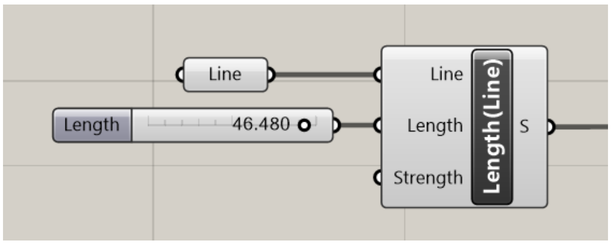


FIGURE 4.25: Length block

Coincident goal: this goal maintains the input parameters coincident. This block accepts the two points as input [Fig. 4.26].

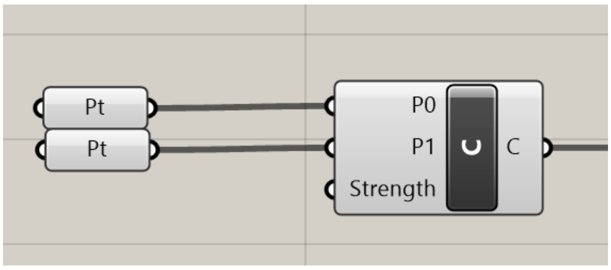


FIGURE 4.26: Coincident block

Angle goal: this goal keeps two segments at a predefined relative angle to from each other. The segments can be either disconnected or have a shared vertex. This block accepts as input the two segments as LineA and LineB; and the angle, in radians, between the segments. As for the Length block, the angle input can be set by external parameters, a numerical 1D slider, or by the position of the controller sent via UDP. In both cases, an intermediate block that converts the numerical data from degree into radians is mandatory [Fig. 4.27].

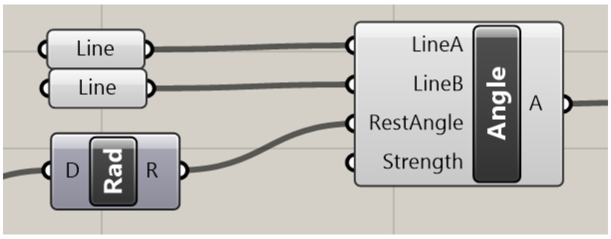


FIGURE 4.27: Angle block

4.3.3 Constraints definition

In the following section, we will describe the Kangaroo diagrams of the joints implemented in the system. The diagrams have to be modeled in advanced and they represent the high – level rules necessary to assemble components in Unity. The input geometry of the kangaroo blocks is selected in Unity through the input device, for example HTC Vive controllers, and then the text string data in sent via UDP.

Hinge

Once the deconstruction of the B – Rep of the geometries is performed, the hinge joint required the definition of two couples of mating points, highlighted in green in the block diagram and in evidence in the Rhinoceros viewport [Fig. 4.28].

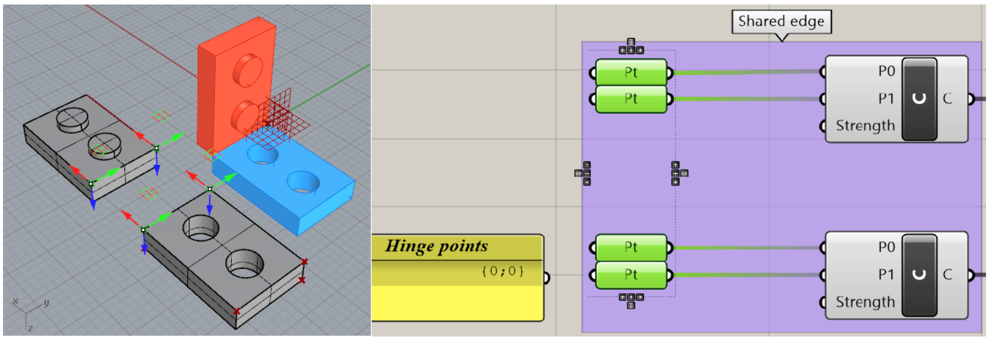


FIGURE 4.28: Selection of the point of the hinge

Then it is necessary to set the opening angle between the two components. This operation is performed via the Angle goal. Two edges on the longest side of the components, highlighted in green in [Fig. 4.29] are selected as LineA and LineB input of the Angle goal component. The angle in radians is set in Unity accordingly to the Euler angles of the input device implemented, and the angle value is sent via UDP to Grasshopper.

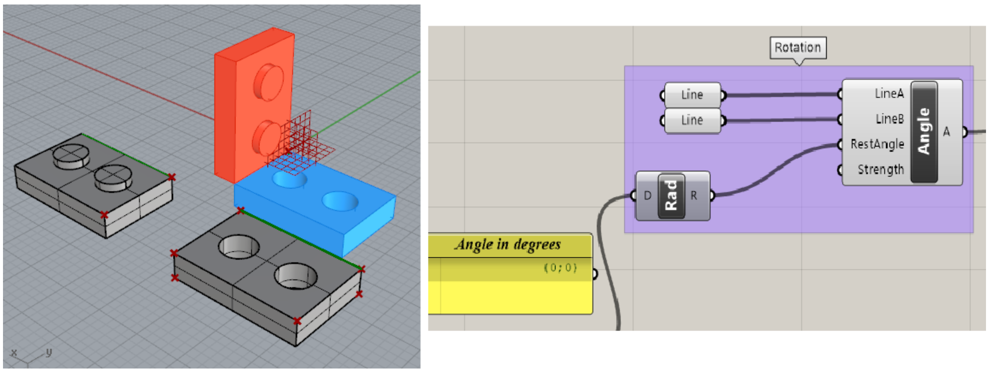


FIGURE 4.29: Selection of the rotating angle of the hinge

Lastly, one of the components is set as frame component by locking all the Degrees of Freedom via three OnPLane blocks for three sets of points which are locked to the XY, XZ and YZ planes in the origin O = (0,0) [Fig. 4.30].

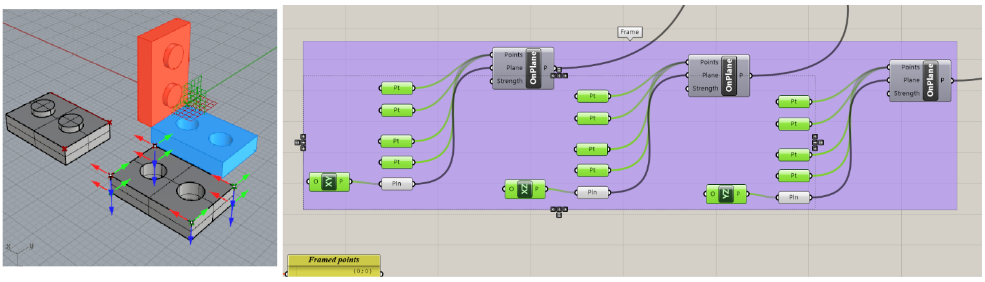


FIGURE 4.30: Framing of one of the two components

Aligned faces

In order to align two faces of two rigid bodies, the AlignFaces block is required. This block needs four input in order to create a coupling. The block requires the framed planes P, output of the two rigid bodies, and the two faces to be aligned. With the deconstruction block of Grasshopper, the input geometries are deconstructed in their components and the user can select the two faces to align. Thanks to the OnPlane block previously described, one of the rigid bodies can be fixed and set as frame. Lastly, a Coincidence block is used in order to match two vertices of the faces [Fig. 4.31].

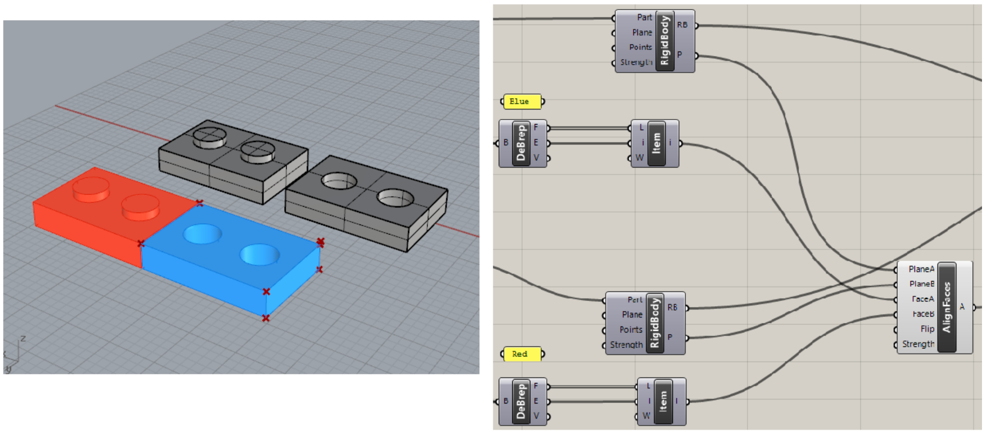


FIGURE 4.31: Alignment of two faces

Pin/hole

In order to create a pin/hole joint, it is necessary to set the coincidence constraint of the pin axis with the hole axis. Hence, the Concentric block is used. The Concentric block accepts as input the axis of the RigidBodies to be aligned and the framed planes of the RigidBodies to be assemble [Fig. 4.32].

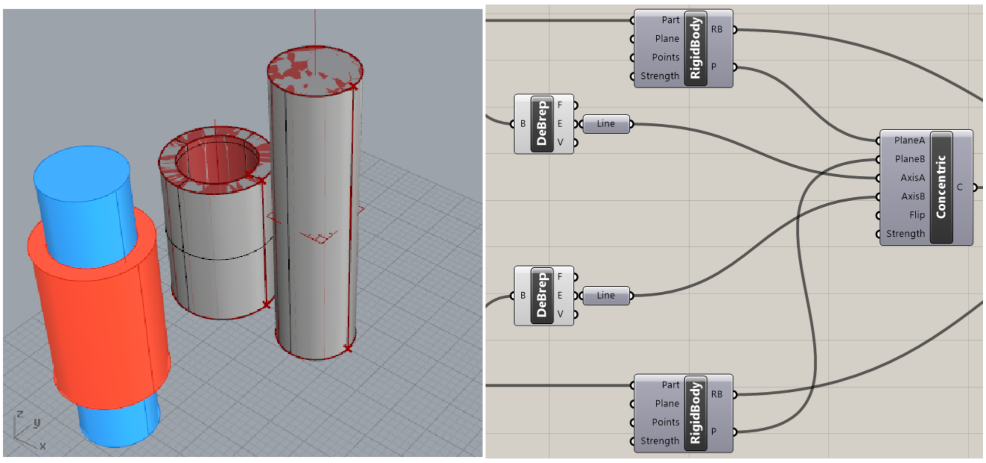


FIGURE 4.32: Creation of a pin/hole constraint

Once the alignment of the axis is accomplished, the pin is framed by fixing the position of the points on the cylinder base to the selected plane [Fig. 4.33]. All the input for selecting the geometries are sent from Unity to Grasshopper via UDP.

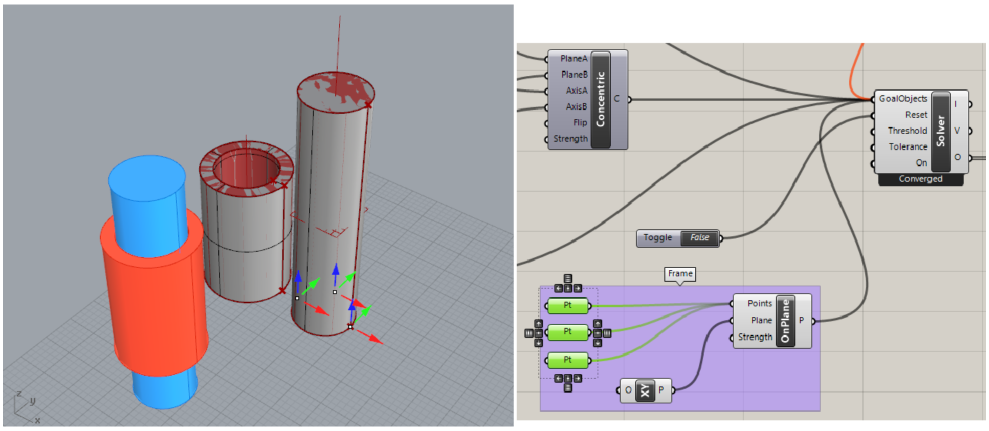


FIGURE 4.33: Framing of the red component

Slider

The sliding coupling is created by the combination of the CoLinear block and the ClampLength block. The Colinear block allows to align a set of points, while the ClampLength block allows to define a trajectory along which the slider is enabled to move [Fig. 4.34].

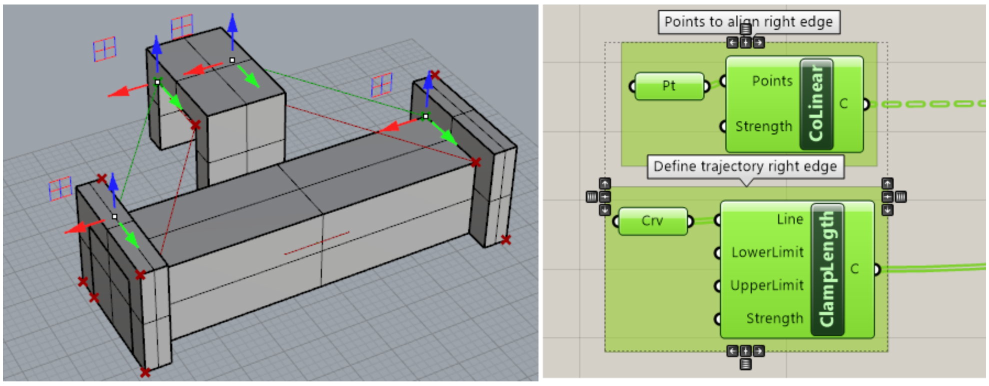


FIGURE 4.34: Definition of the sliding trajectory for the upper edge

The previously described steps are duplicated for the other edge of the slider joint [Fig. 4.35]

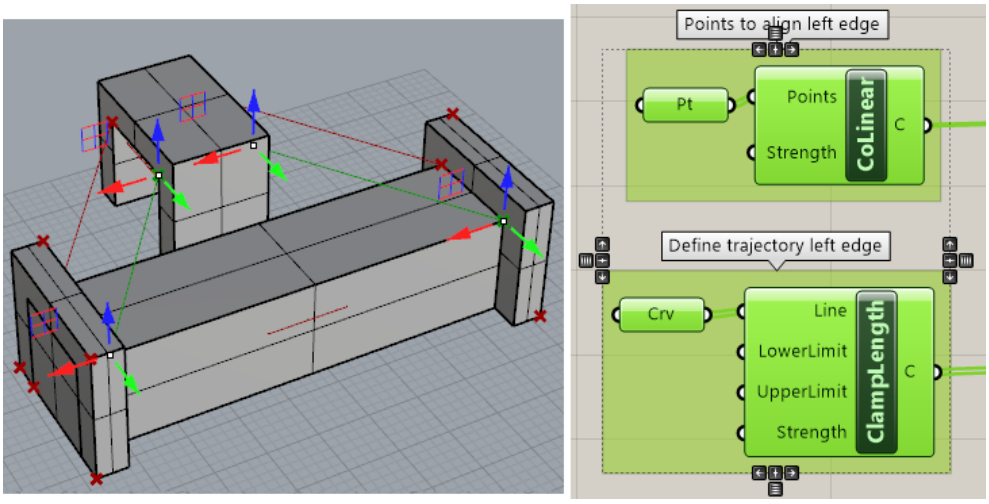


FIGURE 4.35: Definition of the sliding trajectory for the lower edge

Eventually, the guide component is set as frame by using the OnPlane block and all the six Degrees of Freedom are locked. The final coupling of the guide component and the slider component is depicted in [Fig. 4.36].

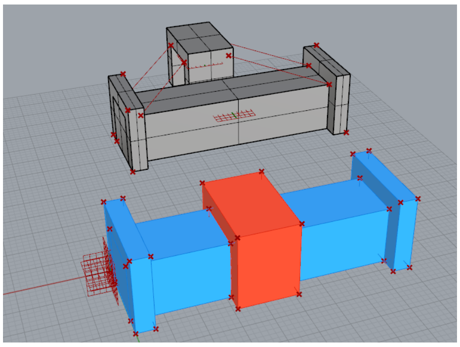


FIGURE 4.36: Final assembly of a slider

4.3.4 Conclusion

In this chapter we described the implementation of a modeling environment within the Unity3D game engine. The modeling environment is based on the ProBuilder API and it is both a tool of creation of new meshes and an editing tool for imported meshes, thanks to the ProBuilderize classes and methods which convert imported meshes in editable meshes. The Unity modeling environment is connected via UDP to Rhinoceros's Grasshopper and the input of the immersive hardware are collected within Unity and sent to Grasshopper in order to exploit the Grasshopper high – level editing capabilities for geometries modifications. Besides the modeling features, an assembly modeling tool has been implemented by using Kangaroo components and the input for selecting mating elements, such as vertices, edges or faces, are received via UDP by the input device. A set of diagrams for basic joints has been presented, comprehending the input and the output of each block.

Chapter 5

Training in Immersive Environments

In this chapter we describe various applications of the immersive environment for the simulation of the assembly and disassembly sequence. More specifically, the first application is an Augmented Reality (AR) application and it consists in the simulation of a disassembly of a remote controller. The research work has been presented at the 27th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM) in 2017. The second application is a Virtual Reality (VR) application and it is the result of the research conducted at the Oregon State University. It consists of a simulated learning environment for the timber construction field. The third application is a Mixed Reality (MR) application in the medical context and it is part of a wider research project that aims to develop new tools for the pre – operative surgical planning and surgeons' training and it has been presented for the application in the international project MAECI: Call for joint research projects Italy - USA (2019-2021) by the Italian Minister of Foreign Affairs and International Cooperation.

5.1 Application #1: Immersive Technologies for disassembly sequence optimization and workforce training

5.1.1 Research background

The environmental problems require adopting better disassembly procedures at the product's End of Life. Planning innovative disassembly strategies in the Early Stage of the Design (ESD) process can improve the development of sustainable products with an easier decommissioning and a more recycling – oriented approach. Nowadays many Computer Aided Process Planning software and optimization algorithms provide optimized assembly or disassembly sequences, but they are mainly based on a time and cost compression approach, neglecting the human factor. By leveraging the use of AR, the user has the possibility to effectively interact with the real object while implementing new disassembly sequences, suggested by the optimization algorithm. The novelty of this approach relies in the fact that the augmented environment is not only a testing environment for assembly sequence or training, but the user experience may become the input as new parameters in the optimization process. In such a way, the data exchange between virtual environments and real world is enhanced and intangible elements, like human experience, can be digitalized and become part of the company know – how. Based on the modeling system previously described, this research aimed to develop a tool for the assessment of the assembly and disassembly strategy. In particular, the research focused on the disassembly sequence planning and it is called Augmented Reality Disassembly Evaluation Tool

(ARDET). By using ARDET it is possible to place real – life 3D objects in the working area in front of the operator. The advantage of ARDET use during the ESD is to ameliorate the Decision Making (DM) process of the disassembly planning: thanks to the ARDET implementation, new original disassembly strategies can be discovered based on the direct users' interaction with the virtual product. Furthermore, user can simulate and practice with the disassembly sequence in the first stages of the product development process, when only 3D models are available. Hence, operators will be fully operative and productive when the assembly lines are ready. The integration of an immersive disassembly sequence planning tool with a disassembly sequence optimization algorithm enhances the selection of the sequence as it can be modified in an immersive and interactive manner by the user if the sequence proposed by the optimization algorithm is unfeasible or requires cumbersome operations for the worker. In [Fig. 5.1] it is depicted the comparison workflow between the traditional workflow for the sequence planning, without the use of immersive technologies, and with the workflow for the sequence planning that leverages AR capabilities.

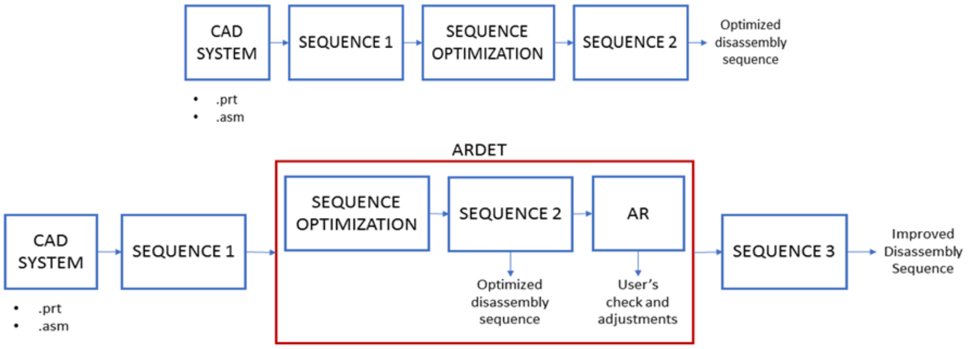


FIGURE 5.1: ARDET workflow

5.1.2 System architecture

ARDET has been developed in two versions, both of them composed by three modules: Image Acquisition Module (IAM); 3D Modeling Module (3DMM); and User Interaction Module (UIM).

5.1.3 ARDET V1

In ARDET V1 [Fig. 5.2] the IAM was based on the ALVAR VTT AR library [43].

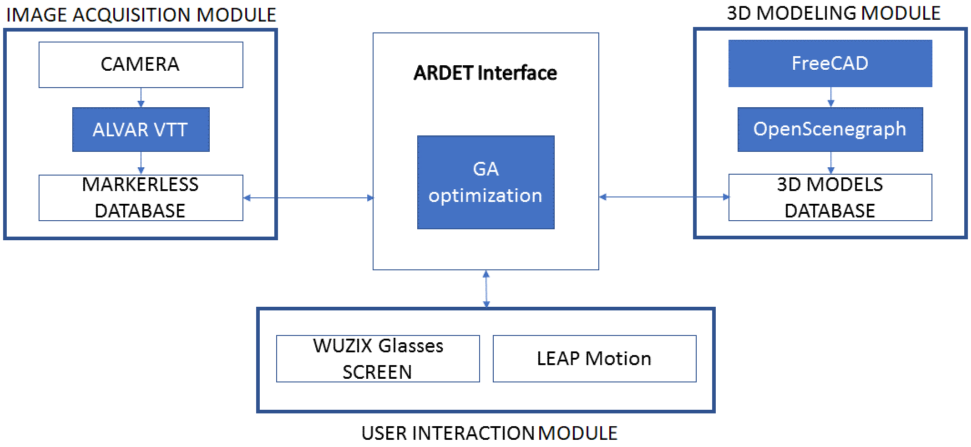


FIGURE 5.2: ARDET V1 implementation

ALVAR VTT is a suite of products for creating AR applications developed by VTT Technical Research Centre of Finland. It supports markerless tracking, based on point clouds created from photos of the component taken from different positions and angles. Then, the point cloud is stored in the database: the image of the real component is than compared to the database to obtain its position in the ARDET reference system. The first step of the acquisition procedure is the training of a Fern classifier with a component picture. The component picture must be in the range of 200 x 200 up to 500 x 500 pixels and it should contain a number of unique features sufficient to train the pose and position estimation algorithm. Once the training is complete, the points cloud is stored in the markerless database and it can be used for tracking and superimposing the 3D model on the real object in real time. In [Fig. 5.3] it is shown the unique features detected in the picture of a remote controller for an air conditioner by ALVAR during a training session.

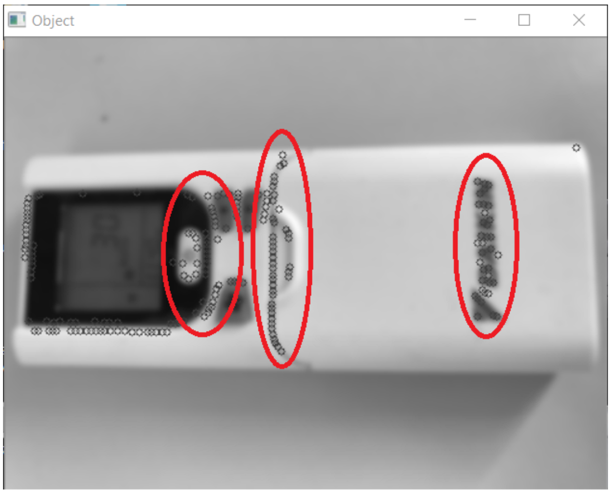


FIGURE 5.3: FERN training

While in [Fig. 5.4] presents the recognition of the same features on a picture of the controller taken in real time from a user-set point of view. It is worth to note that the comparison of the key point distances in the database pictures and in the video streaming allows to ALVAR to detect the camera point of view in terms of positions and angles in an absolute and camera reference systems.

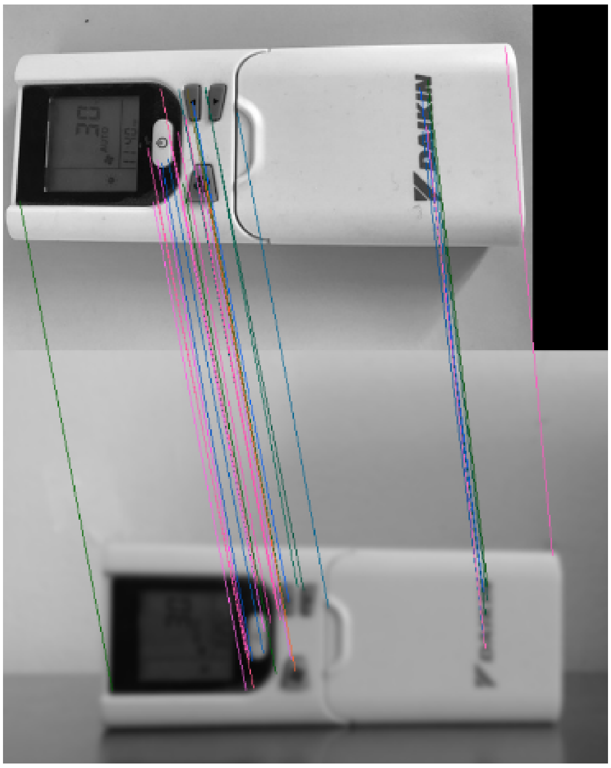
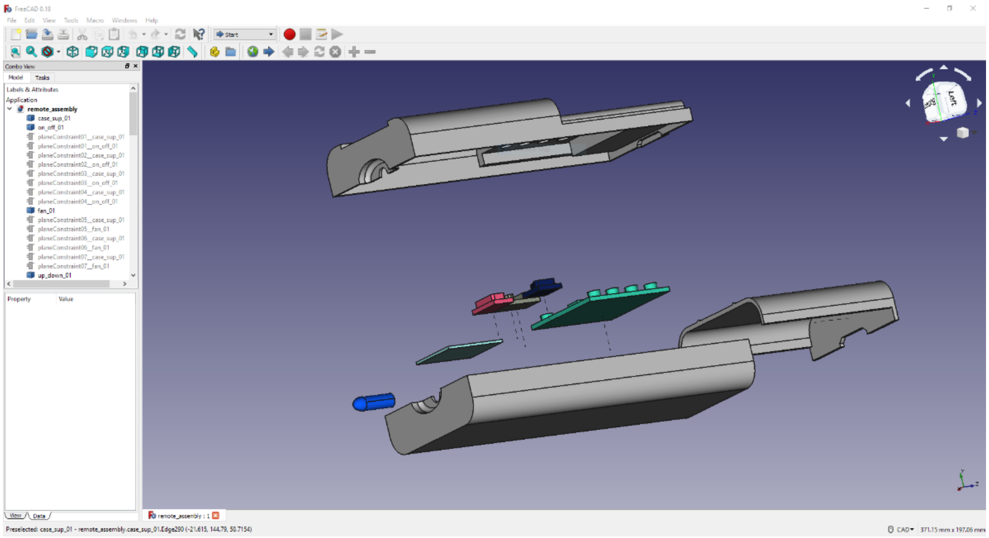


FIGURE 5.4: FERN training

The 3DMM is based upon the open source FreeCAD 0.16 3D modelling software. FreeCAD has been chosen due to its easy personalization capabilities. FreeCAD is based upon a set of workbenches, each one specialized to perform a task among with parts modelling, assembly, and sketching. New user-developed workbenches can be implemented with Python code and a lot of them is freely available on the Internet. Following the ARDET procedures, once the parts of the product to study has been modeled and assembled, it is necessary to define the disassembly trajectories. These can be derived from the Genetic Algorithm Optimization (GAO) process. In order to do this, a specific workbench of FreeCAD has been used, named Exploded Assembly. This workbench allows to visualize the disassembly trajectories of every component and it allows to show the movement required to disassembly the component. These trajectories can be seen in [Fig. 5.5] as dashed lines.


FreeCAD 0.18
File
Edit
View
Tools
Macro
Windows
Help
Start
Combo View
Model
Tasks
Labels & Attributes
Application
remote_assembly
case_sup_01
on_off_01
planeConstraint01__case_sup_01
planeConstraint01__on_off_01
planeConstraint02__case_sup_01
planeConstraint02__on_off_01
planeConstraint03__case_sup_01
planeConstraint03__on_off_01
planeConstraint04__case_sup_01
planeConstraint04__on_off_01
fan_01
planeConstraint05__case_sup_01
planeConstraint05__fan_01
planeConstraint06__case_sup_01
planeConstraint06__fan_01
planeConstraint07__case_sup_01
planeConstraint07__fan_01
up_down_01
Property
Value
View
Data
remote_assembly : 1
CAD
371.15 mm x 197.06 mm

Preselected: case_sup_01 - remote_assembly.case_sup_01.Edge290 (-21.615, 144.79, 58.7154)

FIGURE 5.5: Trajectories generation in FreeCAD

After the modeling of the product 3D model in FreeCAD, it is than necessary to convert it to .OSG format, the OpenScenegraph native format. This step is necessary because ALVAR VTT supports only the OpenScenegraph format. Osgconv, a free plugin of OpenScenegraph, has been used to convert the model from the FreeCAD native format to .OSG format. All the models are then stored in the ARDET 3D models database. In order to make ARDET more adequate for the disassembly operations, a see – through HMD has been adopted. This solution provides immediate heads up access to information that the user would normally access by looking down at a hand-held display or a screen, thus losing the contact with the real product being under the disassembly process. The Vuzix Star 1200 XL Glasses has been used in this study. LEAP Motion Controller device has been integrated in the ARDET platform to achieve a real-time user interaction with the 3D model. Leap Motion allows hand tracking and gesture recognition in a conic space centered in the LEAP Motion with a diameter of 0.4m, and a height of 0.4m. As noticed by [42] the best performances are obtained with palms facing down hands at a distance of 0.2m from the LEAP. A preliminary set up is necessary in order to define the tracking sensibility, focusing on finding a compromise between precision and usability. The interaction with the model is obtained by a pinch gesture made with fingers. While maintaining the pinch, the user can translate and rotate each component in accordance with the disassembly path. If fingers are released, the interaction ends and the selected component remains in the last tracked position. The scene has been set up in Unity. Vuzix Glasses and Leap Motion have been chosen because in both cases a Unity package can be downloaded from the web. Hence, the setup of the virtual scene is easier and less time consuming.

5.1.4 ARDET V2

In ARDET V2 the implementation of Microsoft HoloLens as HMD required some changes in the system architecture concerning the IAM and the UIM. The upgrades regarded the object tracking library and the input commands used for the application. Modifications have been implemented even in the 3DMM since ProBuilder and Grasshopper's Kangaroo have been integrated in ARDET, thanks to the UDP exchange data workflow previously described. The architecture of ARDET V2 is depicted in [Fig. 5.6]

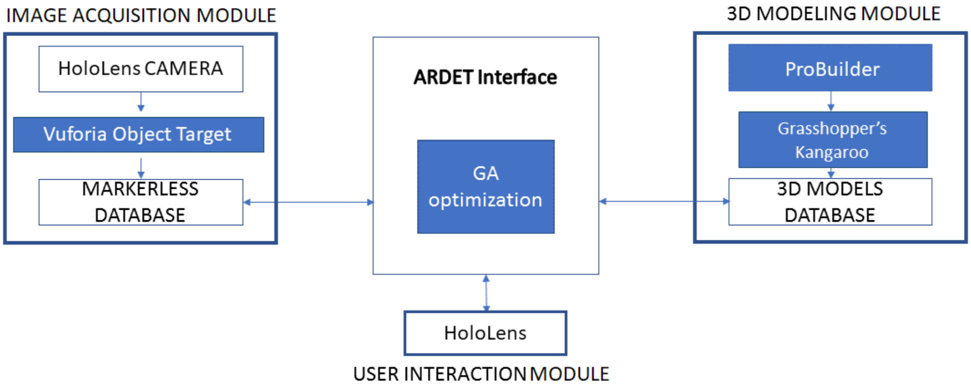


FIGURE 5.6: ARDET V2 implementation

The IAM is based on Vuforia SDK. The point cloud creation procedure is similar to the one performed with ALVAR VTT library and both Vuforia and ALVAR VTT support markerless tracking. The difference consists in the fact that ALVAR VTT detects points of interest on a 2D image, while Vuforia allows to detect points on a real 3D object by using the Object Target. To create an Object Target, the Android app Vuforia Object scanner is required. In order to scan the object, the specific Vuforia greyscale marker is necessary and the user has to pose the object in the scanning area [Fig. 5.7].

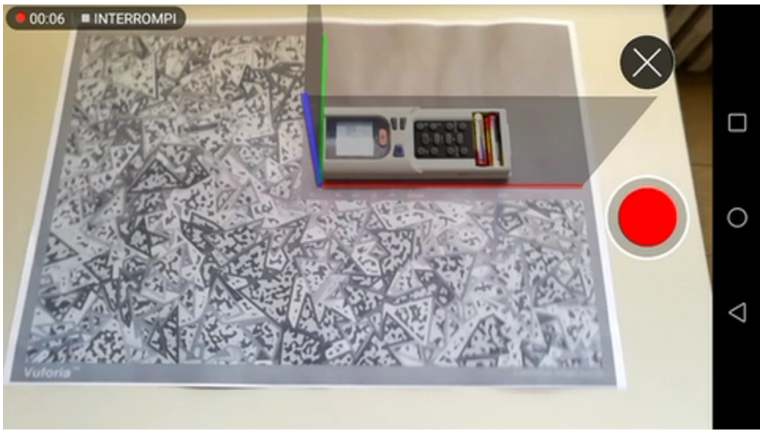


FIGURE 5.7: Object scanning procedure with Vuforia object scanner

Then, the application converts the images in greyscale and starts detecting the points of interest on the object. A dome is depicted in the UI of the App and the dome is divided in several sectors. The application informs the user on the number of points detected and a minimum of three points for each sector is required in order to successfully track the object position. As the scanning proceeds, the scanned sectors become green, meaning that the sector has enough points of interest on it. When the majority of sectors has been scanned, i.e. almost all the dome sectors are green, the scanning is completed and the point cloud dataset is uploaded to the Vuforia Developer Portal [Fig. 5.8]. Then, the user can download the UnityPackage containing the dataset and imports it inside the Unity Project.

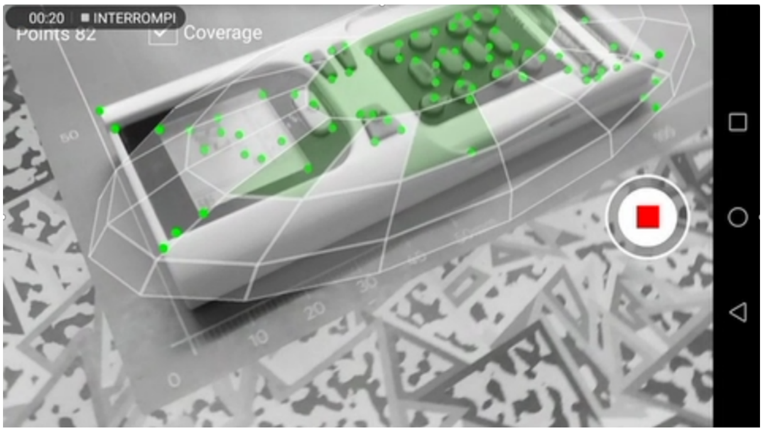


FIGURE 5.8: Point cloud creation

Thanks to the implementation of HoloLens, the UIM has been simplified. In ARDET V1, the UIM was composed by the Vuzix glasses as HMD, and Leap Motion controller, as hand tracking device. In ARDET V2, Microsoft HoloLens integrates both the functionalities, since hands tracking capabilities are embedded in HoloLens and these functions can be used in C# scripts, by calling the corresponding methods and classes contained in the HoloToolkit by Microsoft. In the ARDET version implementing HoloLens it has also been developed a training functionality with voice command capabilities. Through the voice commands, the user can select between two different modes: disassembly evaluation mode and disassembly training mode [Fig. 5.9]. Then, he can navigate through the disassembly sequence by using play, pause and next voice commands. This training function has been made possible in ARDET V2 because HoloLens implements voice commands, while Vuzix glasses did not.

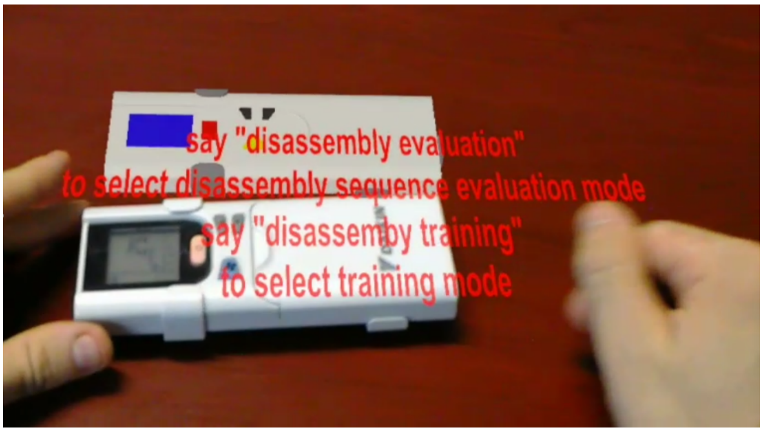

say "disassembly evaluation"
to select disassembly sequence evaluation mode
say "disassemby training"
to select training mode

FIGURE 5.9: Disassembly evaluation and training in AR

5.1.5 Disassembly optimization workflow

ARDET algorithm for optimization of the disassembly sequence is a matrix – based decision making approach and it follows several steps. It is based on product geometry analysis, components relationships, joints classification and unmounting movements. The steps are:

1. Training session and creation of the target to track the object.

2. Computation of the assembly relationships with the method prosed by [44]. According to the assembly constraints, a Disassembly Strategy Matrix (DSM) is automatically composed. The DSM comprehends the mutual moving directions of each component. Each matrix value represents the mutual Degree of Freedom (DOF) of each couple of components present in the assembly. If contact between two components does not exist, the matrix position of these two components is filled with a zero;

3. The Genetic Algorithm (GA) optimizes the Disassembly Strategy (DS) once the DMS is created. The proposed DS is the sequence from which the user can explore different disassembly solutions, based on his skills and experience.

4. Thanks to the DSM, the presence of a removable component is checked. The values in the DMS represents the direction along x, y, z axis reference system along which the disassembly is possible. The first component that must be disassembled is the one which presents the lower number with non-zero values in the DMS matrix correspondent row. When two or more values in the DSM matrix are different from zero, the part is removable only if the values are similar (e.g. +x and +x).

5. According to [44], if a removable component exists, the algorithm assigns a score in order to define the component disassembly precedence. The score is given according to four parameters: the first one represents the number of subassemblies that are unconstrained after the given disassembly operation. The second parameter corresponds to the number of tool changes: precedence is given to the operation not requiring tool changing. Third parameter concerns product handling: precedence is given to the operation not requiring product repositioning. Fourth parameter evaluates the overall easiness of the disassembly task with the combined use of two scores: one ergonomic, obtained from standard tables of ergonomics; one visibility score, derived from the worker optimal visual area [45]. The two stated scores are then combined in order to form the fourth parameter, named accessibility score.

6. If all the matrix positions are equal to zero no component one is removable, so that the disassembly process has been finished.

5.1.6 ARDET case study

The ARDET tool has been tested in several case studies, one of which is reported in this section. The object selected for this example of disassembly procedure planning is a remote control composed of nine parts: upper case; lower case; sliding cover; LCD display; transmitter's infrared light; buttons (see Figure 6). The user starts the disassembly simulation wearing the HMD and frames the remote control. The Vuforia library recognizes the object previously scanned as a target in order to superimpose the corresponding 3D model upon the real object. Once the disassembly session is started, ARDET suggests each disassembly operation to accomplish the entire disassembly task. The operator can simply follow the GA optimized sequence or eventually he can create new disassembly sequences, in accordance with instructions of the DSM. In fact, the DSM points out the subsequent possible removable component for each disassembling step: for example, after removing the sliding cover, the operator can decide whether to remove the upper case or the lower case. In this example, the upper case removal gets a higher score compared to the lower case removal. Indeed, the second operation demands the remote-control repositioning, in accordance with the ARDET algorithm. The disassembly sequence score is updated in real time after sequence changes made by the operator and the new score can be compared with the GA optimized sequence scoring. In this way, the operator immediately realizes whether the change he wants to implement introduces some ergonomic improvements, maintaining a score similar to the optimal one.

Conclusion

The ARDET platform could be considered as a new tool to support assembly and disassembly strategy design and evaluation. The use of ARDET allows to superimpose real – life 3D model upon the real object and it permits the visualization of information concerning sequence path and/or specific tools required for the accomplishment of the desired task. The first benefit deriving from the use or AR is a greater interactivity in the design of the disassembly sequence planning. Thanks to ARDET, the designer can immediately test the ergonomics and easiness of the different operations. Another advantage derived from the adoption of the tool is the possibility of introducing the worker disassembly experience in the sequence optimization process. In this way, the expertise and know-how relating to the disassembly operations are actively exploited in the definition of the unmounting path and disassembly sequence. The integration of the user know – how in the automated optimization criteria for the disassembly planning ameliorates and upgrades the disassembly process design. Notably, the implementation of AR tools in industrial scenarios is subjected to some difficulties due to low/weak textured objects with shiny surfaces: these features can generate inaccurate 3D model pose; furthermore, the images definition and accuracy depends on the lighting conditions of the environment: this limitation of the use of AR tool in industrial scenarios can be overcome by using more advanced devices, like Microsoft Hololens, which can better manage sudden changes of lighting conditions. Another aspect to be considered is the user's acceptance: the hand gestures set implemented in ARDET is quite limited and the fingers and hands movements are not so natural and intuitive. Even with Microsoft HoloLens, hand tracking gestures are not easy to perform and the 3D models movements are not so precise. For these reasons, ARDET implementation in industrial context requires a preliminary training session for the user. Moreover, the use of HMD can be cumbersome, due to the weight of the device, and space limited, due to the necessity for Microsoft HoloLens of WI-FI connection with the computer. Furthermore, future works could involve making ARDET interacting with a commercial CAD software, in order to test the platform in a more plausible industrial scenario.

5.2 Application #2: A simulated learning environment for AEC workforce training and manual skills development

5.2.1 Research background

Wooden Light – Frame (WLF) structures offer flexible design possibilities for civil engineering fields in North America. Their flexible assembly procedures rely on a blend of standardized operations and well-developed best practices that expert construction workers acquire during their educations and careers. Standardized technical knowledge and career expertise guarantee high quality and low operation times. According to [46], construction field is going to be one of the most growing sector by 2026. In the next few years, however, construction sector will encounter a shortage of the expert workforce due to the retirement of a large part of baby – boomers currently employed in the construction field [47]. In order to face the imminent shortage of expert workforce, new teaching techniques must be investigated in order to foster the education of new construction workers and capitalize on the workers' expertise. The purpose of this research is to design, develop, and evaluate a simulated learning environment that improves users' understanding of installation, erection, and construction processes with WLF structures using VR. The proposed system will allow users to engage with a VR representation of a small building site. It will render timber studs, panels, and carpenter's tools while allowing the user to assemble and disassemble the WLF structure in accordance with predefined connections. Concurrently, users may cycle between discrete phases of installation from start to finish. The proposed system aims to be a useful tool both for teaching construction techniques and for training workers' manual skills. Through this research, we will evaluate the effectiveness of training in VR for manual skills development in the wood construction field. The assessment is based on the comparison of the proposed VR training system with the traditional training used by the construction company. The long – term outcome of this research will be the production of a comprehensive study of VR training for different personnel involved in the construction industry; namely architects, engineers, and builders/installers. Optimization and realistic virtual simulation of the installation process are relevant research subjects in the construction field in order to guarantee a seamless turnover for the future of expert workers in the construction field, while maintaining the highest standard of quality.

5.2.2 Methodology

The methodology we implemented comprises of several phases. In the first phase, each trainee group takes a preliminary knowledge test (pre-test) concerning technical knowledge about WLF components and installation techniques. The second phase is either a VR or a traditional training, which is divided in three steps. Lastly, each group takes a System Usability Scale (SUS) questionnaire to evaluate their perceptions of the system. The workflow is depicted in [Fig. 5.10]

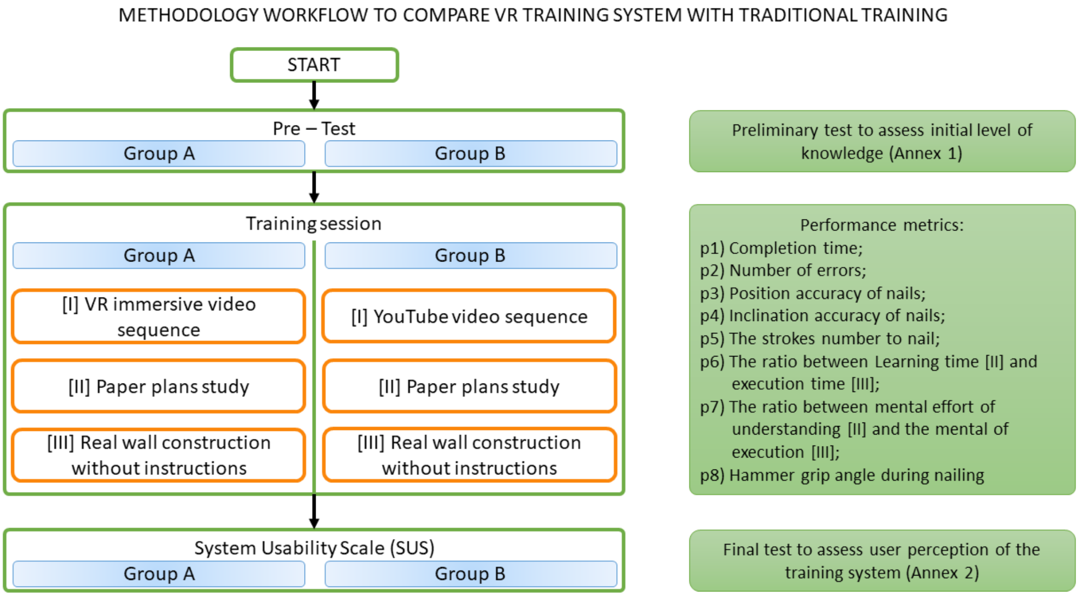

METHODOLOGY WORKFLOW TO COMPARE VR TRAINING SYSTEM WITH TRADITIONAL TRAINING
START
Pre – Test
Group A
Group B
Preliminary test to assess initial level of knowledge (Annex 1)
Training session
Group A
Group B
[I] VR immersive video sequence
[I] YouTube video sequence
[II] Paper plans study
[II] Paper plans study
[III] Real wall construction without instructions
[III] Real wall construction without instructions
Performance metrics:
p1) Completion time;
p2) Number of errors;
p3) Position accuracy of nails;
p4) Inclination accuracy of nails;
p5) The strokes number to nail;
p6) The ratio between Learning time [II] and execution time [III];
p7) The ratio between mental effort of understanding [II] and the mental of execution [III];
p8) Hammer grip angle during nailing
System Usability Scale (SUS)
Group A
Group B
Final test to assess user perception of the training system (Annex 2)

FIGURE 5.10: Training methods comparison workflow

5.2.3 Preliminary test

According to [48], we designed a preliminary test to assess the knowledge background of every participant. This preliminary test is fundamental because it determines what the trainee already knows, which can be used to determine what the trainee learns from the subsequent training session. Since during the training session, the trainee takes part to a complete WLF construction procedure which contains five main tasks, the preliminary test consists of ten questions, two questions for each task.

5.2.4 Training session

The training sessions contained three steps, as previously depicted in [Fig. Oregon workflow]. The first step [I] is the video sequence: Group A watched a VR immersive video derived from a series of videos, whereas group B watched the original videos. The series of videos describes the construction sequence of a WLF wall. To create the immersive video, we analyzed the original ones, derived the task analysis, and edited it according to the suggestions of the construction company. In this way, we ensured that our task analysis and construction sequence were as correct as possible and that they reflected the actual construction sequence performed by the workers of the construction company. In the second step [II], participants from both groups can study construction plans of the tasks. The third step [III] is the execution stage. Group A and Group B performed the wall construction in real – life without any instructions. During this stage the performance of the user is evaluated with our performance metrics.

5.2.5 System Usability Scale (SUS)

A 5 – point Likert scale assessment of users' acquisitions of manual skills is also included. The subjective performance test quantifies the technology reliability level that trainees perceive and is important in order to evaluate how much Information Technologies (IT) such as VR are identified as useful and accepted by users in an industrial context, which is especially pertinent since perceived usefulness and ease of use are decisive in the learning process [49]. The subjective performance evaluation is part of the satisfaction survey in order to assess how much VR training is perceived effective and trustworthy by trainees in improving manual work. The SUS assessment is valuable for training upgrades and revisions due to its repeatability and associated literature.

5.2.6 Performance metrics

To compare our VR training system with traditional training, we chose a set of parameters to quantify the performance of the participants. Recent research in the VR assembly training advocate for time and error count as measurements to assess the user performance: For example, in [50] for the industrial mining field and in [51] for the assembly of medical devices [52]. We used completion time and error count because our end goal is to train workers to competitive competence levels, and the lower the completion time and fewer wrong steps, the shorter the construction time of the final product.

5.2.7 Learning content

To compare our VR training system with traditional training, we chose a set of parameters to quantify the performance of the participants. Recent research in the VR assembly training advocate for time and error count as measurements to assess the user performance: For example, in [50] for the industrial mining field and in [51] for the assembly of medical devices [52]. We used completion time and error count because our end goal is to train workers to competitive competence levels, and the lower the completion time and fewer wrong steps, the shorter the construction time of the final product.

5.2.8 WLF learning content

Theoretical knowledge regarding WLFs is transferred through several means. As technical knowledge, simplified 2D technical drawings of WLF structures are shown to the trainee during VR training. 2D representations are fundamental for two reasons. Firstly, they are the only support available on the building site, so every worker has to know how to read and understand a 2D technical drawing. Secondly, 2D technical drawings develop theoretical knowledge about the WLF wall assembly techniques and encourage the maintenance of schematic – acquired knowledge over the long term. Specific terminology is used in 2D technical drawings distinguish different parts of the WLF wall. For example, in a typical WLF wall, different type of studs can be used. Depending on their position and purpose, they have different lengths and designs. “Top plate” and “bottom plate” refer to studs running horizontally, between floors and ceiling. “King studs” are full length central studs which connect the bottom and top plates. “Corner studs” are full-length studs, which are double studs with a series of blocks between them if they form the corner of two perpendicular elements, or single studs they connect walls continuing straight. “Jamb studs” are studs beside the window studs. They are shorter than full-length studs because they directly support the window lintel (the windows architraves). Depending on the timber used and the lintel span, there may be one, two, or three jamb studs underneath the lintel. "Jack studs" are any studs not running from top plate to bottom plate: for example, over the lintel or underneath the sill trimmer (the lower support of the window). Finally, "noggins", or "dwangs", are horizontal reinforcements between full-length studs. A simple WLF wall is depicted in [Fig. 5.11].

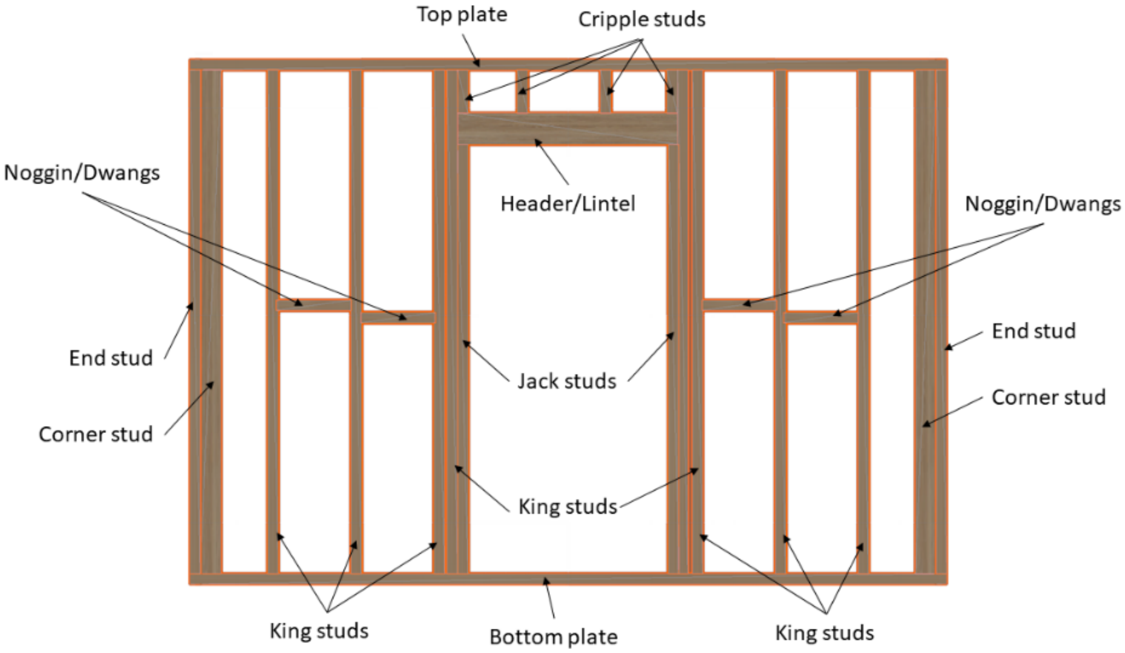


FIGURE 5.11: Wall layout of the case study for the training sequence

5.2.9 WLF wall assembly sequence

The VR training session implements a 3D model assembly sequence instead of a 2D video recording. The 3D model assembly sequence offers two advantages: firstly, it allows the viewer to freely change their point of view, and secondly, the user can create the appearance of the building site, enhancing the application's immersiveness. It also integrates pause and rewind functions, so the trainee may dwell on the more complex passages to better comprehend them [Fig. 5.12].

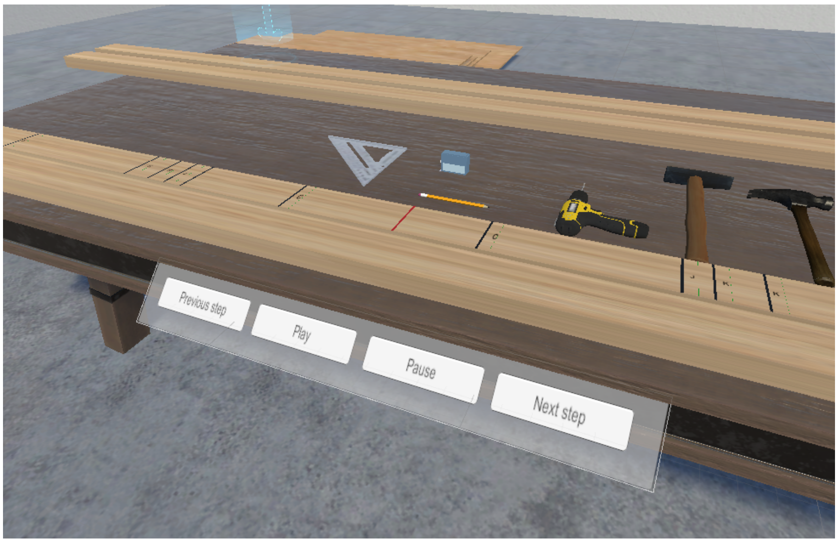
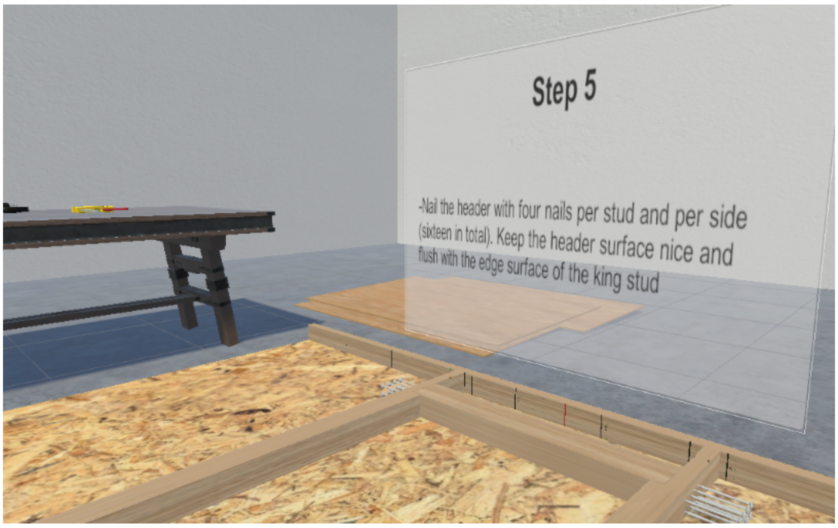


FIGURE 5.12: User interface in VR

Moreover, best practices identified by experienced workers may be combined with the 3D assembly sequence. This would demonstrate the usefulness of specific manual operations. A few examples of best practices are described as follows. Full-length studs must be positioned such that all timber′ crowns are aligned in one direction: this prevents the wall from being wavy when the final front panels are installed. When common studs frame a door or a window, the studs should be as straight as possible in order to avoid misalignments with the doors and windows architraves. Horizontal noggings prevent vertical buckling if they are put in the middle of the studs. They can also prevent lateral-torsional buckling if they are posed at the bottom of the studs. Moreover, noggins are suggested to be cut slightly longer than the space between vertical studs in order to fit tightly to the studs. Before starting to attach panels to the wall, the user has to ensure that the wall section is square by measuring the diagonals of the wall. After this, user can start to attach panels when the wall is on the ground. The use of double top plates is recommended when the wall needs to be connected with other walls: a portion of the upper top plate is cut and then it is replaced by the upper top plate of the other wall, creating an overlapping intersection that gives more stiffness and stability to the entire wall. In the same way, corner intersections are assembled with double top plates and an overlapping area. All this information is described in the VR application by using canvas and panels placed in correspondence of the components to be assembled [Fig. 5.13].



Step 5

-Nail the header with four nails per stud and per side (sixteen in total). Keep the header surface nice and flush with the edge surface of the king stud

FIGURE 5.13: Instruction on a canvas in the VR environment

5.2.10 Carpenters' equipment learning content

The learning content includes 3D models of tools and typical building site equipment. The main functionalities and instructions for tools are described to the user via dedicated pop-up menus that the user can activate. For example, these instructions might describe how to change drill bits while using a power drill or how to recharge a nail gun. Moreover, safety information and risk prevention behaviors in the use of this equipment are described to the user.

5.2.11 Learning outcomes

Learning Outcomes (LO) of any training course must be clear. They are descriptions of what trainees must know and be capable of after the course. In order to describe LOs of the proposed VR training course, we applied the revised Bloom's taxonomy of learning [53]. According to this taxonomy, LOs must be described with specific and measurable terms. Qualitative terminology must be avoided because verbs like "to know", "to understand", "to appreciate" are too vague. Action – type verbs like "to remember", "to list", "to describe" are preferable because they are connected to the level of learning involved. The adopted taxonomy lists six Levels of Learning (LoL) [Fig. 5.14] and it suggests appropriate "action" verbs for every level.

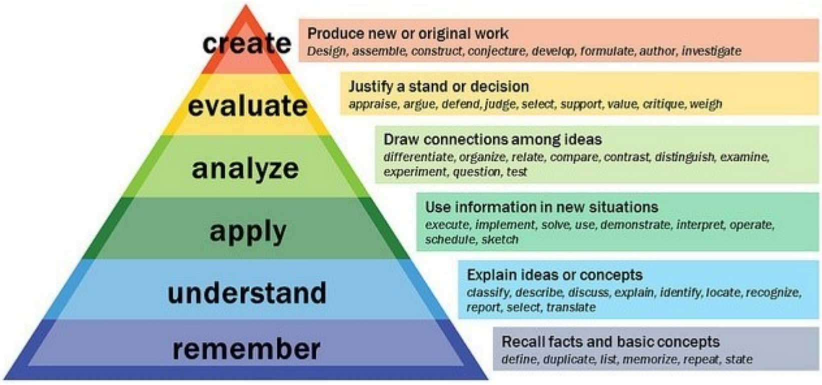

create
Produce new or original work
Design, assemble, construct, conjecture, develop, formulate, author, investigate
evaluate
Justify a stand or decision
appraise, argue, defend, judge, select, support, value, critique, weigh
analyze
Draw connections among ideas
differentiate, organize, relate, compare, contrast, distinguish, examine, experiment, question, test
apply
Use information in new situations
execute, implement, solve, use, demonstrate, interpret, operate, schedule, sketch
understand
Explain ideas or concepts
classify, describe, discuss, explain, identify, locate, recognize, report, select, translate
remember
Recall facts and basic concepts
define, duplicate, list, memorize, repeat, state

FIGURE 5.14: Revised Bloom's taxonomy

For each item in the task analysis and for each LoL, if applicable, we defined LOs according to the skill developed and knowledge acquired by the trainee. Moreover, such LOs facilitate the definition of the pre training questionnaire for the evaluation of the initial level of knowledge of the trainee. The preliminary test before the training session is necessary in order to understand what the trainee learns during the subsequent training session.

5.2.12 Preliminary test

The LOs are arranged so that at least one question can be defined for every LoL. Following the LOs structure described above, we designed a preliminary test to evaluate the initial level of knowledge of the participants. Since the construction sequence consists of five tasks, the preliminary test contains two questions for each task for a total of ten questions. In this way, every task has the same weight with respect to the others. Every question of the preliminary test is a closed – ended question, with yes or no as possible answer. However, if the user answers yes to a particular question, he will have to prove his knowledge with an open-ended answer. In such a way, it is discouraged to answer yes to questions in order not to show lack of knowledge. All the possible questions we devised for the preliminary test are listed in Appendix A. The preliminary test has been developed using Qualtrics, a provider of tools for online survey of the Oregon State University.

5.2.13 Experimental procedure

A between subject experiment has been performed in order to test the validity of the VR training system. The independent variable of the experiment consists in the type of the training: VR training or traditional training, i.e. i.e. watching the operator building the wooden frame for the wall. Since traditional training in a real construction company was difficult to organize for time and safety reasons, the traditional training has been simulated by watching video materials collected from several online source, suggested by the construction company in order to ensure the reliability of the video teaching materials. Due to the complexity of the overall construction sequence, we conducted the experiment only for the first two tasks: Wall Layout and Wall Framing Then both group performed these two tasks of the wall construction in real – life. The dependent variables of the experiment are the completion time of the wall construction sequence and the number of errors made by the participants during the real – life construction. The task analysis previously carried out by the researchers let: firstly, to identify the most crucial activities to be taught and to subdivide the main task in smaller sub – tasks; and secondly, to identify what sub - tasks the user performs correctly in order to define the percentage of the sequence correctly accomplished. With the proposed approach, the researchers can analyze the user experience and compare the final results of the two training session. Time and errors are recorded and used to compare the performance of the two groups. For this experimental comparison, the HTC Vive Pro has been used as HMD and the tasks have been modeled inside Unity3D in different scenes: every task has a specific sequence of animations that replicates the sequence performed in the videos [Fig. 5.15].

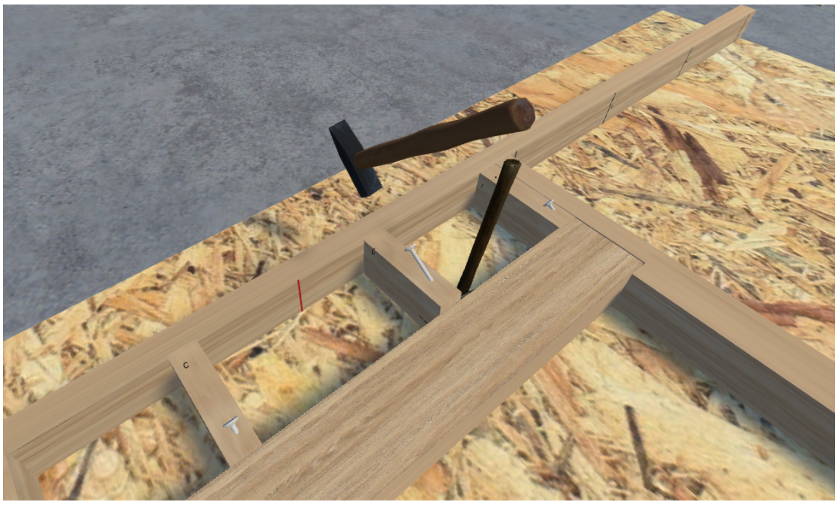


FIGURE 5.15: Animation of a specific step, toe nailing

5.2.14 Participants

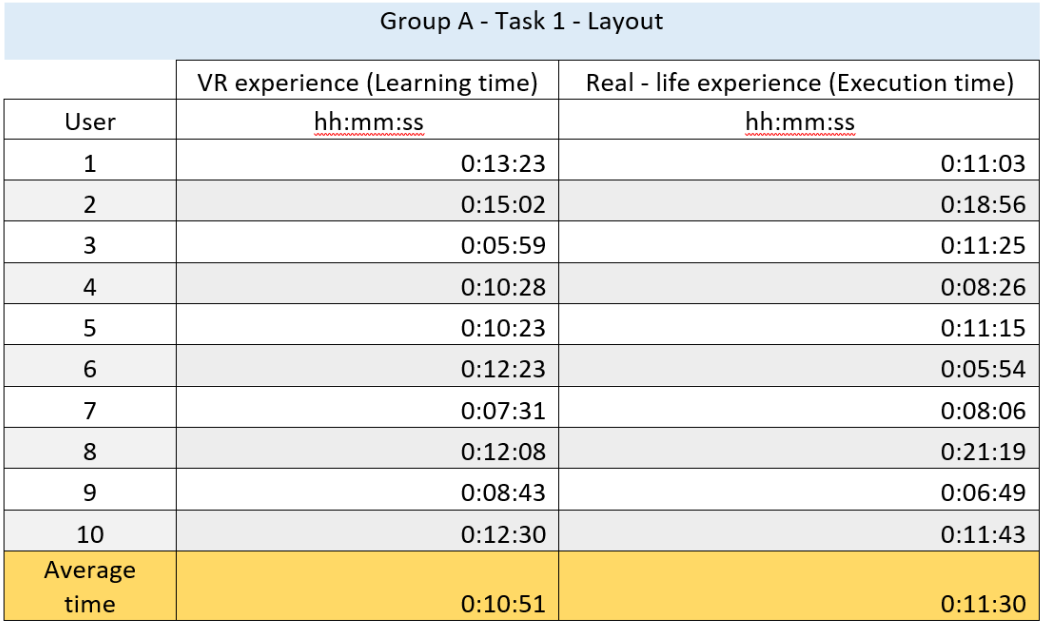
20 students from degrees in Mechanical Engineering and Industrial Product Design of the University of Bologna have been asked to participate in the experiment. The participation was on a voluntary basis and there was any kind of reward for participants. Some of the participants had already experienced immersive technologies, either VR or AR. In order to create a common level of experience among all the participants, a short training session of the HTC Vive has been conducted by the research team and the participants had the time to familiarize with the Vive system: they could walk around in the virtual environment and learn the input of the Vive controller to interact with virtual objects. Participants were divided in two groups; each group is composed by 10 participants with an equal number of male and female participants for each group. The preliminary VR training session and the gender – based partition are necessary to control possibly prior experience and gender effects on the results, since both prior experience and gender can influence the UX when dealing with immersive technologies as studied by [54]. The timber construction sector is mainly a male – dominated sector, with less than 2% of female carpenters and practitioners [55]. In order to allow women to take advantage of the expected growth of this sector in the next few years, we included 4 female participants for each group. The average of the participants is 24 years old. 12 (60%) of the participants are male, and 8 (40%) are female.

5.2.15 Measurements

Three kind of measurements have been taken: errors, completion time and perceived usability of the system. Number of errors and completion time are used to determine the overall user performance. An error is defined as: 1) a step is performed in a different way from how suggested and the result of the step create a different result; 2) a step is missing; 3) incorrect use of the carpenters' tools. Two types of errors have appeared during the experiment: independent error and dependent error. Dependent errors are errors originated by a previously made error. Independent errors are error which are isolated and not related to previously made errors. Perceived usability of the system has been evaluated by the SUS questionnaire at the end of the training.

5.2.16 Results

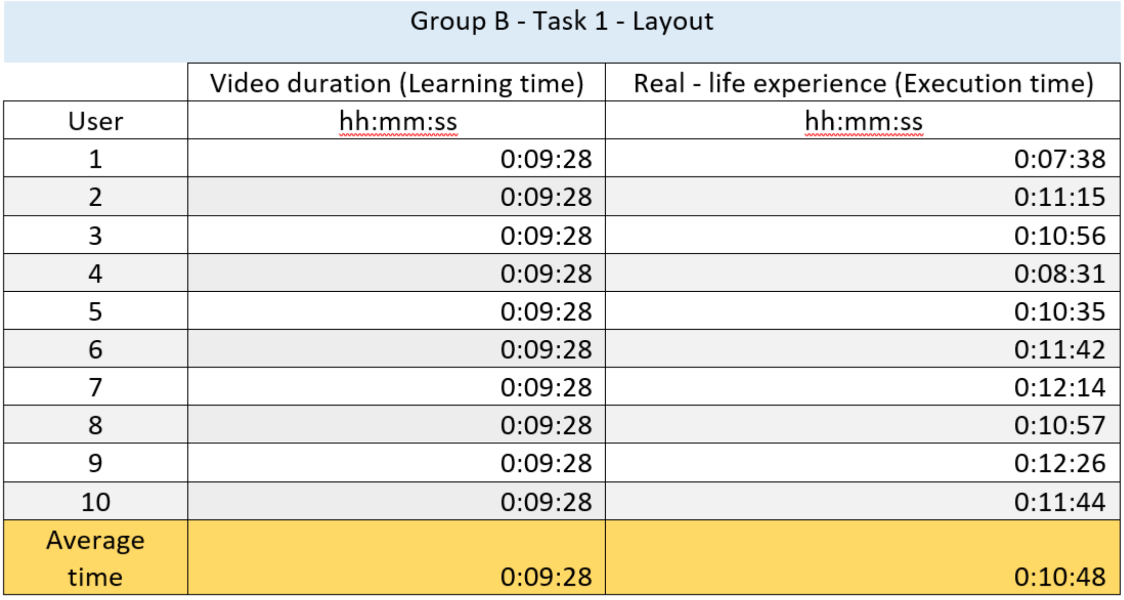
For each task, training time and execution time are reported below. Group A has performed the training in the VR environment, and the time that each user has spent in the VR training environment and time spent in the layout task are reported in Table 1:



Group A - Task 1 - Layout		
	VR experience (Learning time)	Real - life experience (Execution time)
User	hh:mm:ss	hh:mm:ss
1	0:13:23	0:11:03
2	0:15:02	0:18:56
3	0:05:59	0:11:25
4	0:10:28	0:08:26
5	0:10:23	0:11:15
6	0:12:23	0:05:54
7	0:07:31	0:08:06
8	0:12:08	0:21:19
9	0:08:43	0:06:49
10	0:12:30	0:11:43
Average time	0:10:51	0:11:30

Table 1

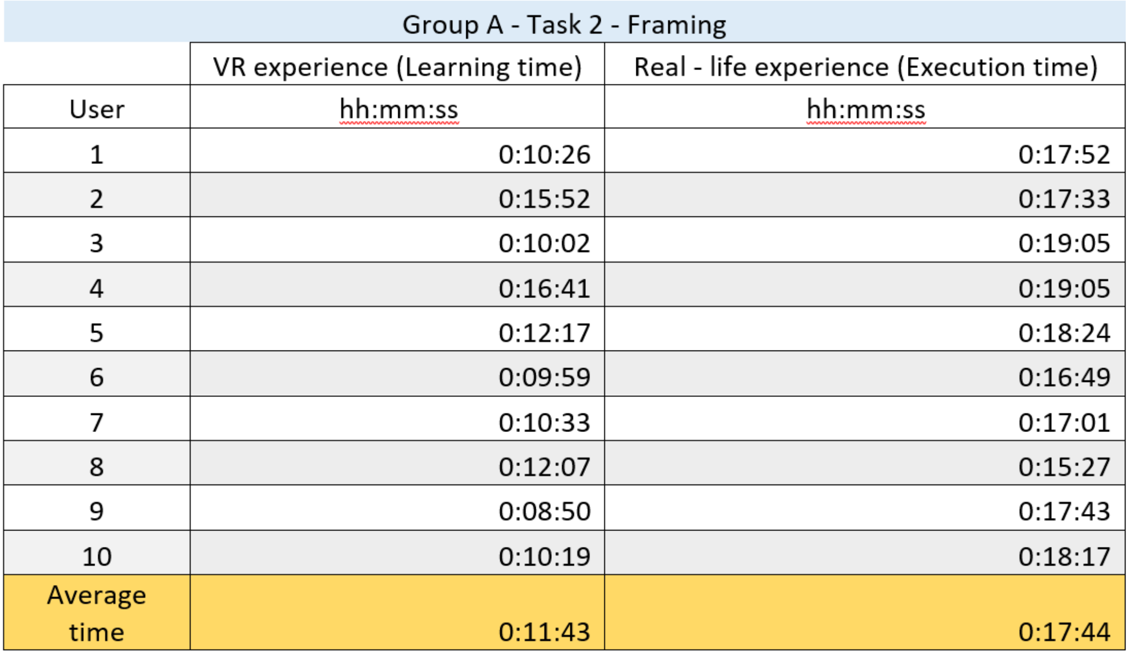
Group B has been trained with video material, hence the learning time is equal for each user. The time that each user has spent in the layout task are reported in Table 2:



Group B - Task 1 - Layout		
	Video duration (Learning time)	Real - life experience (Execution time)
User	hh:mm:ss	hh:mm:ss
1	0:09:28	0:07:38
2	0:09:28	0:11:15
3	0:09:28	0:10:56
4	0:09:28	0:08:31
5	0:09:28	0:10:35
6	0:09:28	0:11:42
7	0:09:28	0:12:14
8	0:09:28	0:10:57
9	0:09:28	0:12:26
10	0:09:28	0:11:44
Average time	0:09:28	0:10:48

Table 2

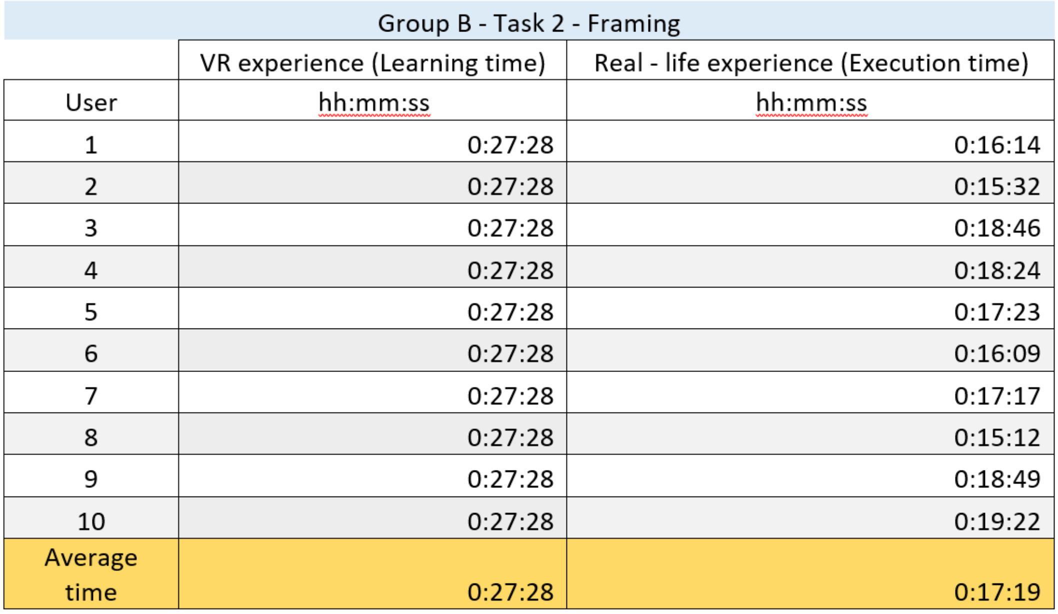
The Group A VR training time and execution time for task 2 are reported in Table 3:



Group A - Task 2 - Framing		
	VR experience (Learning time)	Real - life experience (Execution time)
User	hh:mm:ss	hh:mm:ss
1	0:10:26	0:17:52
2	0:15:52	0:17:33
3	0:10:02	0:19:05
4	0:16:41	0:19:05
5	0:12:17	0:18:24
6	0:09:59	0:16:49
7	0:10:33	0:17:01
8	0:12:07	0:15:27
9	0:08:50	0:17:43
10	0:10:19	0:18:17
Average time	0:11:43	0:17:44

Table 3

The Group B video training time and execution time for task 2 are reported in Table 4:



Group B - Task 2 - Framing		
	VR experience (Learning time)	Real - life experience (Execution time)
User	hh:mm:ss	hh:mm:ss
1	0:27:28	0:16:14
2	0:27:28	0:15:32
3	0:27:28	0:18:46
4	0:27:28	0:18:24
5	0:27:28	0:17:23
6	0:27:28	0:16:09
7	0:27:28	0:17:17
8	0:27:28	0:15:12
9	0:27:28	0:18:49
10	0:27:28	0:19:22
Average time	0:27:28	0:17:19

Table 4

It is particular relevant the comparison of the average total training time for the first two tasks with respect to the average total execution time. In the VR set – up, the average training time is 22 minutes and 34 seconds; while in the traditional set – up, the average training time is 36 minutes and 56 seconds. With both training method, the average execution time is approximately 29 minutes. This means that, with respect to the execution time, VR training is more effective in term of time, hence, cost for the company. Another significant comparison is the comparison of the errors. With VR training the users made in total 20 errors during real life construction, while with traditional training users made in total 26 errors. Hence, the average errors of VR training are 2 errors per user, while for traditional training is 2,6. Furthermore, with both training methods, one of the participants did not finished the construction sequence. The difference between these two errors is that the participant trained with traditional method stopped performing construction in step 3 of task 1 "layout"; while the participant trained in VR stopped construction in step 1 of task 2 "framing". This means that VR training performed slightly better than traditional training. Even with respect to the average total time of training, the VR training performs better since it generates less errors than traditional training in the construction stage. It is worth noting that the majority of the errors made by participants are related to the presence of a human being during the training. Three errors occurred most frequently: the wrong application of the "16 on center" rule; the incorrect use of the speed square; and the incorrect crowning procedure. The reasons why these errors occurred more frequently in the VR set – up is that in the VR set – up a VR trainer is not present, hence, the trainees did not see how to perform a specific operation or to use a tool. Regarding the "16 on center" rule, in the video the worker showed to hook the tape to the bottom plate edge and started measuring from the edge in a continuous way. However, in the VR environment the tape is moving by its own from one of the bottom plate edge along the studs [Fig. 5.16].

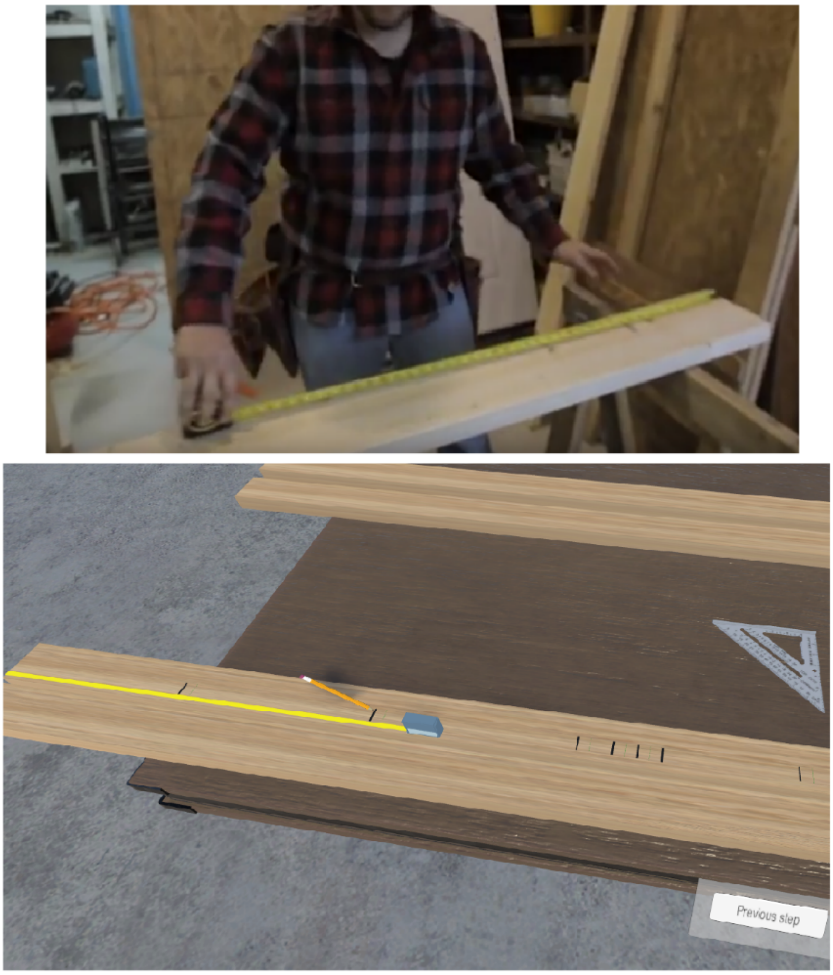


FIGURE 5.16: The measuring step in the video training mode and in the VR training mode

Some of the participants did not hook the tape to the bottom plate edge, rather they measured 16 inches every time from the mark they previously draw [Fig. 5.17].

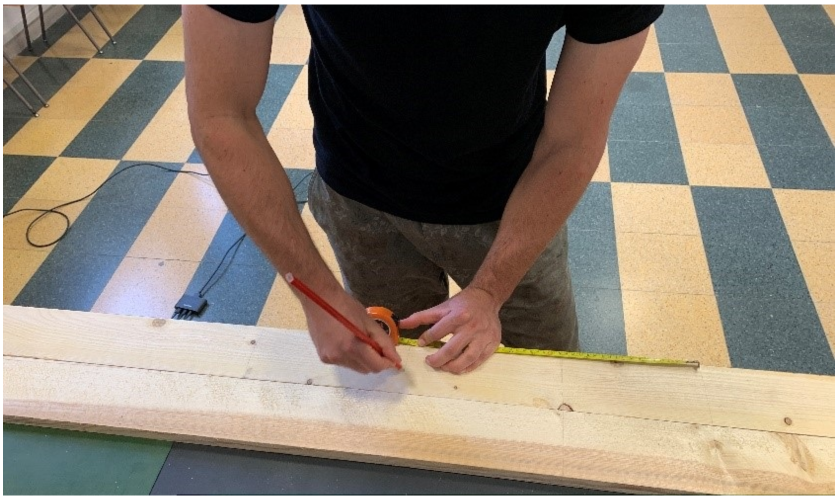


FIGURE 5.17: Incorrect use of the tape after VR training

Regarding the handling of the speed square, all the instructions about how the user should handle tools and equipment are written instructions; while, as evidenced in the comments sections of the SUS questionnaire, seeing a human being handling the carpenters' tools could facilitate learning how to use the tools. Despite this suggestion, most of the user of the VR training set – up (6 participants) used the speed square in the correct way [Fig. 5.18].

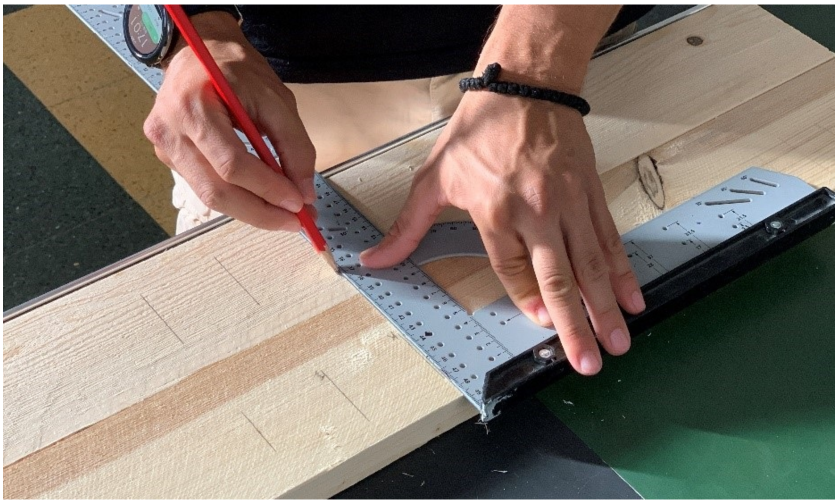
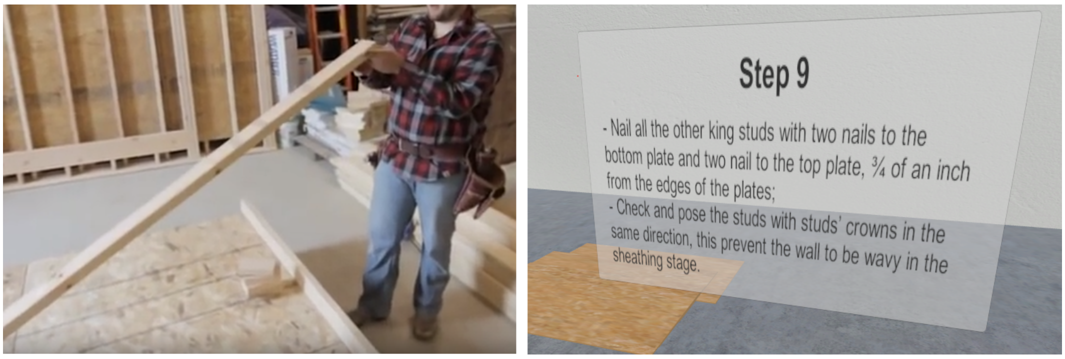


FIGURE 5.18: Correct use of the speed square

Similar to the previous two errors, the crowning step generated some criticalities in the construction sequence. The reason why this step was difficult is that in the VR training environment there was no VR worker showing how to understand the stud crown direction and how to pose all the studs with crowns in the same direction. All the instructions to perform crowning are contained in the canvas and the trainees had to read them without seeing the actual operation. While, trainees that performed the traditional training made fewer errors in this step. In fact, this error occurred 4 time in the VR set – up, and only 3 in the traditional set – up. The difference in how to perform the crowning step is depicted in [Fig. 5.19], on the left in the video and on the right in the VR environment.



Step 9

- Nail all the other king studs with two nails to the bottom plate and two nail to the top plate, ¾ of an inch from the edges of the plates;
- Check and pose the studs with studs' crowns in the same direction, this prevent the wall to be wavy in the sheathing stage.

FIGURE 5.19: Visual instruction by the trainer in the video training mode, and the correspondent instructions on canvas in VR training mode

In [Fig. 5.20], it is depicted a trainee that followed the traditional training performing the crowning operation as suggested by the worker in the video.

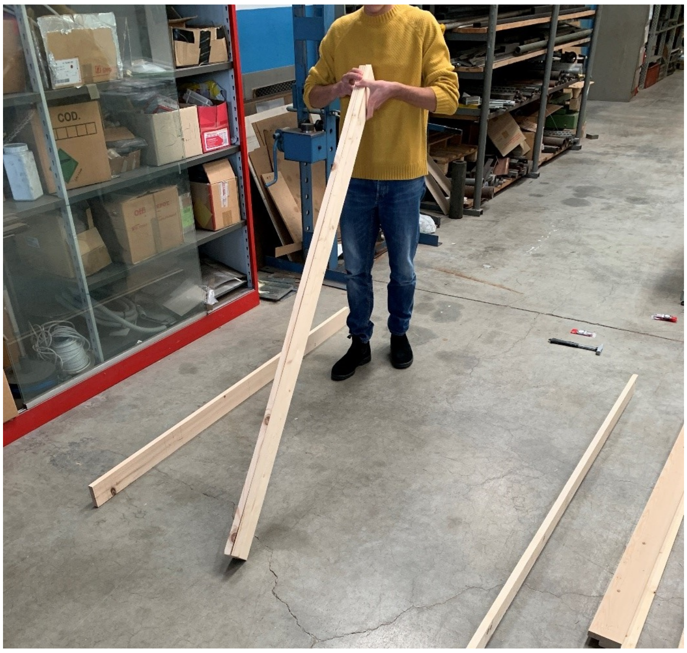
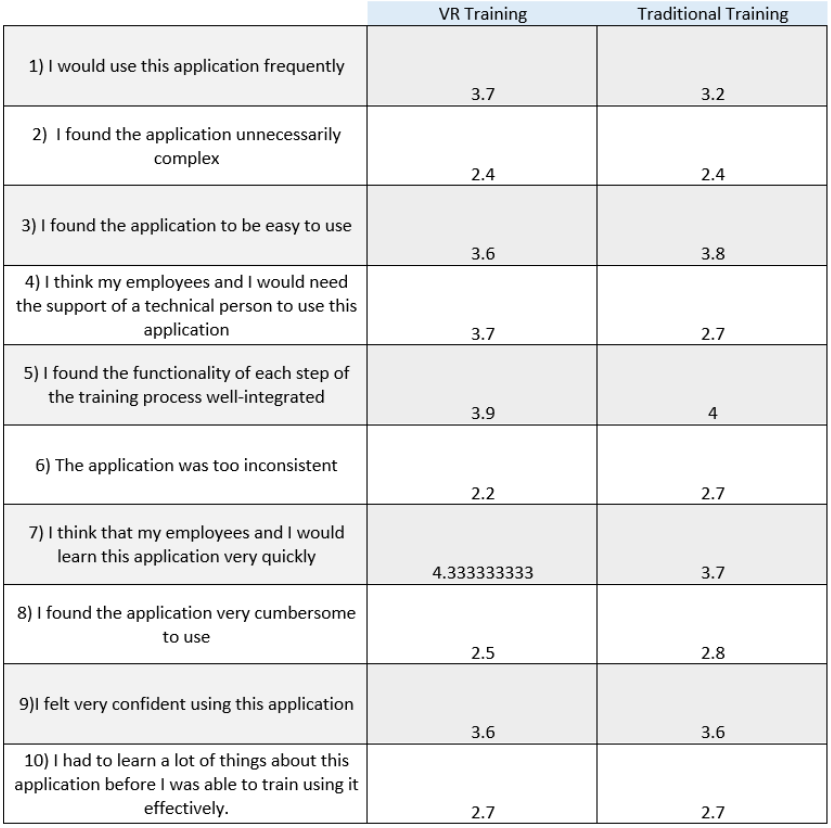


FIGURE 5.20: Correct execution of the crowning operation

In order to better compare the two training systems, it has been added an extra questionnaire after the training session. This second test is called "control test" and contains the same questions of the preliminary test. Participants are requested to answer again to the questions and in this way we could calculate the learning rate of the two training system. Before the training, either VR or traditional, participants did not answer to the questions and all the answer were blank. The reason why this happened is that timber construction is not so usual and it has become more common in the last few years. However, after the training many participants answered correctly to the control test questions. A comparison of the answer of the control test Is necessary to determine the better learning rate of the two training method. After the training in VR, 9 participants out of 10 remembered at least one of the names of the studs used in the framing process. After traditional training, 6 participants out of 10 remembered at least one of the names of the studs used in the framing process. It is worth noting seven types of studs are used in the framing process and with VR training 3 participants remembered all the seven studs' names; while with traditional training none of the participants remembered all the seven names. With traditional training, only two participants remembered up to six names of the studs. After VR training 6 participants could tell the difference between king studs and jack studs, while after traditional training only 5 participants could tell the difference between these two studs. After VR training the "16 on center" rule has been learned by 7 participants, while after traditional training only 6 participants could enunciate the rule. The question regarding the crowns of the studs received more correct answer in the traditional training, accordingly to the previously described issue of the crowning step. In conclusion, the participants that took the VR training answered correctly to the questions of the control test with a higher percentage than participants who too traditional training. Hence, learning rate of VR training is slightly higher, but we can relate this result with the training times of both methods: since



	VR Training	Traditional Training
1) I would use this application frequently	3.7	3.2
2) I found the application unnecessarily complex	2.4	2.4
3) I found the application to be easy to use	3.6	3.8
4) I think my employees and I would need the support of a technical person to use this application	3.7	2.7
5) I found the functionality of each step of the training process well-integrated	3.9	4
6) The application was too inconsistent	2.2	2.7
7) I think that my employees and I would learn this application very quickly	4.333333333	3.7
8) I found the application very cumbersome to use	2.5	2.8
9)I felt very confident using this application	3.6	3.6
10) I had to learn a lot of things about this application before I was able to train using it effectively.	2.7	2.7

Table 5

training in VR environment is 30% faster than traditional training this means that VR training is more effective than traditional training. In the table below [Table 5], the results of the SUS questionnaire are reported:

It is worth noting that VR training is well accepted by the users since in question n.1 “I would use this application frequently”, VR training obtained a higher score than traditional video training. The easiness of the use of the application between VR training and traditional video training is comparable. Despite the acknowledgement of the need for support in using the VR hardware represented by the VR training score in question n. 4, the users also think that the system learning curve is pretty quickly. Although these result could be affected by a small “wow effect”, they also highlight the good acceptance level of immersive technologies compared to the traditional or video training. In the comments section, 4 users suggested to insert voice instructions in addition to tools animations and panels with written instructions; while 3 users asked for a VR trainer who shows how to perform the most difficult operations, in order to add a visual representation of the written instructions.

5.2.17 Conclusion

The performed experiment has been conducted to study and compare a VR training system with a traditional training system for AEC sector. In particular, the system aims to develop manual skills in the next generation of carpenters. Research on VR training systems for manual skills development is not largely performed, as stated by [56], especially in AEC sector. In the AEC sector the VR training systems developed in recent years deal prevalently with safety procedures and large machineries operations, like cranes and forklifts. Hence, this work represents one of the first example of this research. The results showed that the VR training is faster by 30% than traditional training and it gives the same results in term of errors. So that, user can replicate the training several times and increased their performances, with small or no additional costs for the company. The results of this study will be compared with future experiments. In fact, further research on the use of a VR trainer in the VR training system and the implementation of voice instructions should be carried out in order to test the effectiveness of the suggestions of the participants. Furthermore, future research will focus on the influence of complexity of the construction tasks, for example the performance of the user after the VR training in the framing of a wall with a window or framing operations for the roof of the house, which are more difficult operations than the simple wall case study we carried out. Further implementations of the system would involve the development of the wall construction stage in the VR environment in order to compare a training completely performed in VR with a training completely performed in real – life. Lastly, studies on different system of interaction should be carried out. For example, interactive systems based on gloves with sensors, which are becoming available for HTC Vive. These kind of hardware could be useful to implement haptic feedback during the VR training. Hence, the VR training could develop muscular memory in future workers in a more effective way.

5.3 Application #3: A Mixed Reality workflow for pre – operative surgical planning and surgeons training

5.3.1 Research background

The implementation of immersive technologies in the medical sector is particularly interesting and fruitful since doctors can already exploit techniques of high – resolution imaging for depicting and reconstructing human body. In the past years, there have been several attempts to superimpose the information derived from these imaging techniques on patient body. Now, new devices, like Microsoft HoloLens, can perform the overlapping of virtual models and patient in a more seamless way. The system we implemented leverages Augmented Reality and 3D printing technologies and it is part of a wider project that aims to introduce Industry 4.0 tools in the medical sector. The project has been carried out in collaboration with Rizzoli Orthopedic Institute of Bologna, in particular with the Department of Pediatric Orthopedic Surgery. Surgeon precision and a meticulous pre – operative planning are decisive elements in orthopedic pediatric surgery where the difficulties of operations are heightened due to reduced dimensions of patient anatomy. Cutting – edge technologies and innovative visualization techniques can have a significantly positive impact on the medical field, improving the pathology assessment and reducing the invasiveness of surgical operations. In this work, we describe the implementation of a Mixed Reality set – up for pre – operative surgical planning and surgeon training for complex and risky operations. The objective of the research is beyond the simple augmented visualization of 3D model over the patient body, but aims to simulate the cutting operation of the 3D printed patient bone, reconstructed from Computed Tomography (CT) scan of the patient. The application is restricted to bones because tracking of the soft tissues is hard to perform since deformations have to be taken into account. Moreover, tracking of bony structures can be achieved by using specific reference points on patient body.

5.3.2 CT conversion workflow

The workflow for the conversion of patient CT scans to 3D models ready to be visualized in Microsoft HoloLens is depicted in [Fig. 5.21].

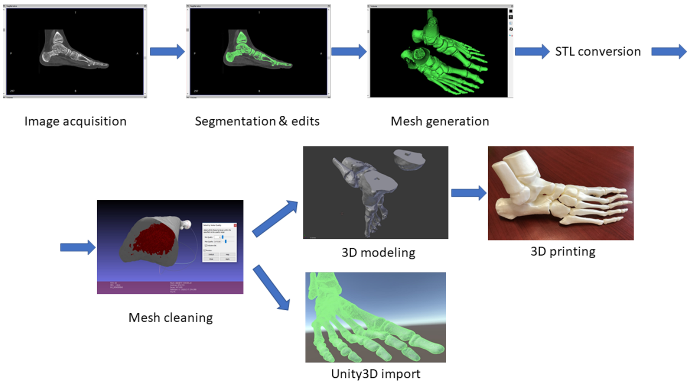


FIGURE 5.21: Conversion workflow from CT scans to 3D models

The CT scan has been performed with patients in the position that minimized the deformation of soft tissues. The 3D data of the CT scans are imported in a segmentation software (InVesalius 3.1) in digital imaging and communications in medicine (DICOM) format. By using density settings, it is possible to visualize different tissues of the body. Setting the appropriate density value allows to isolate the bony tissue and generate the 3D model of the affected area of the patient. InVesalius supports an automatic segmentation and 3D model generation process but the automatic segmentation may not guarantee the generation of a 3D printable volume [Fig. 5.22, 5.23].

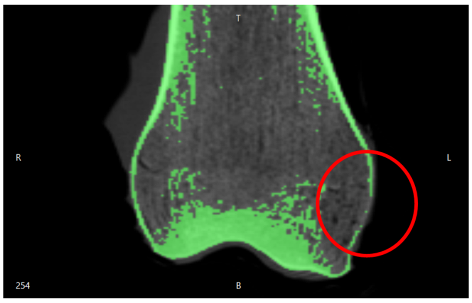


FIGURE 5.22: Segmentation error

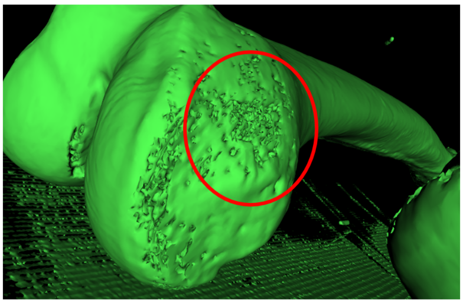
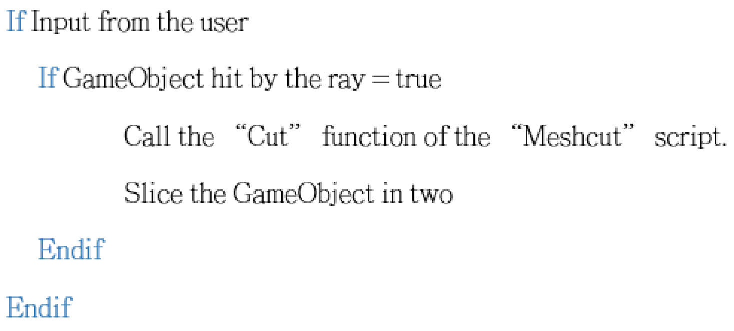


FIGURE 5.23: Incomplete reconstruction

Due to the large amount of scanned images generated by the CT scan to be processed, a manual segmentation process is not a viable solution. To overcome these limitations, a semi – automatic approach has been implemented, where the majority of the bone reconstruction is performed by InVesalius built – in automatic algorithms and then small refinements are made by the user using manual painting functionality of InVesalius. The resulting mesh from InVesalius edits is exported in .stl format and then is further modified in MeshLab 2016. The main operation performed in MeshLab are cleaning and repairing the mesh from errors and irregularities derived from the automatic segmentation algorithms of InVesalius. Mesh cleaning filters are applied in order to remove the internal structures of the model and create a completely empty hollow model. The internal mesh removal has been performed by applying the “Ambient Occlusion” filter, which is a shading technique that calculate how much a point is illuminated by ambient light. As a result of the application of this filter is that the internal triangles are darkened. By applying a threshold value, it is possible to select and delete all the internal triangles. Finally, remaining holes can be filled and the surface in Blender 2.79. All these operations resulted in an 3D anatomical model of the patient bone which is ready for the two following steps: 3D printing of the model used as target in the AR application; and the import of the 3D model into Unity3D.

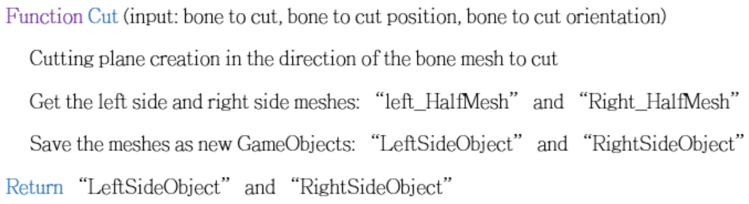
5.3.3 Mesh cutting operations in Unity3D

In order to perform the cutting operation of the bone model in Unity we implemented a MeshCutting C# script. This script leverages the RaycastHit and RaycastAll methods of Unity Physic class, which allow to project a ray colliding with a GameObject, i.e. the mesh of the bone. The pseudocode is described below:



```
If Input from the user
  If GameObject hit by the ray = true
        Call the “Cut” function of the “Meshcut” script.
        Slice the GameObject in two
  Endif
Endif
```

Then, the “Cut” function is the function that perform the actual cutting of the mesh. The “Cut” function accepts as fundamental input the GameObject to cut and the position and orientation of the GameObject. After identifying the GameObject to cut, the function creates a plane in the direction of the GameObject that will be used as cutting plane. Then, for each triangles of the mesh it is computed the UV position with respect to the cutting plane and two arrays are filled with the vertices of the triangles on the left side and on the right side. These becomes the “left_HalfMesh” and “Right_HalfMesh”. Then these are saved as new GameObject and returned out of the function. The pseudocode is described below.



```
Function Cut (input: bone to cut, bone to cut position, bone to cut orientation)
  Cutting plane creation in the direction of the bone mesh to cut
  Get the left side and right side meshes: “left_HalfMesh” and “Right_HalfMesh”
  Save the meshes as new GameObjects: “LeftSideObject” and “RightSideObject”
Return “LeftSideObject” and “RightSideObject”
```

The result of this steps are depicted in the figure below [Fig. 5.24] for a cube which has been divided in three pieces [Fig. 5.25].

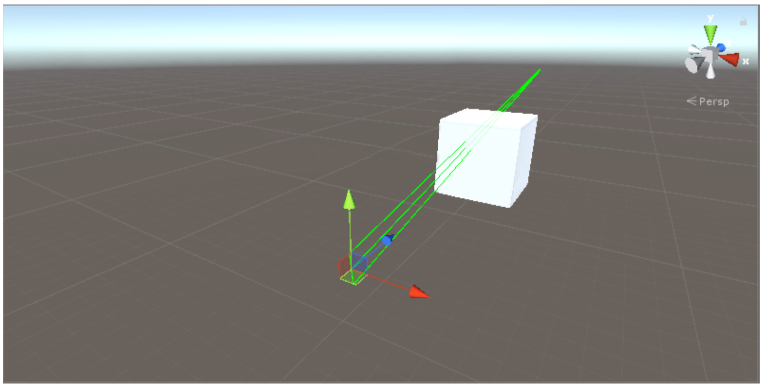


FIGURE 5.24: Raycast of the blade

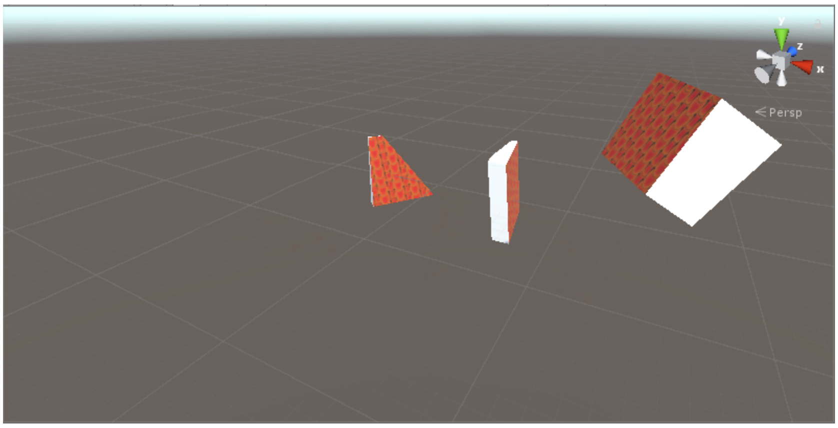
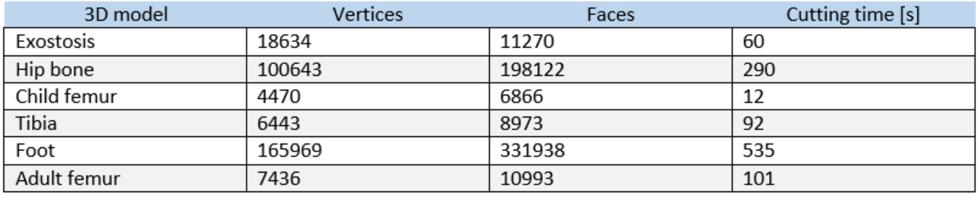


FIGURE 5.25: Sliced object

In order to make available both half part for subsequent operations, such as more cuttings or manipulations, a Collider component is attached to every sub – part in which the original GameObject has been cut. In particular, a MeshCollider has been attached in order to perfectly fit the mesh. The MeshCollider has been set as "Convex" in order to create a closed volume surrounding the mesh.

5.3.4 Cutting complex meshes

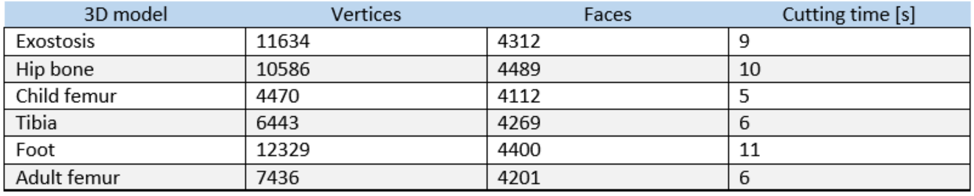
In order to guarantee that the application could run in real – time even with complex meshes, a series of tests has been performed with more complex meshes of bones reconstructed with InVesalius. The cutting operations on these kind of 3D models is computational heavy and the tests we carried out highlighted the unsuitability of the fully detailed meshes from InVesalius for real – time response. In some cases, the cutting time required minutes and this is an unacceptable for immersive applications. In Table 6 are reported the results of the test performed with the most common bones operated by the Rizzoli surgeons.



3D model	Vertices	Faces	Cutting time [s]
Exostosis	18634	11270	60
Hip bone	100643	198122	290
Child femur	4470	6866	12
Tibia	6443	8973	92
Foot	165969	331938	535
Adult femur	7436	10993	101

Table 6

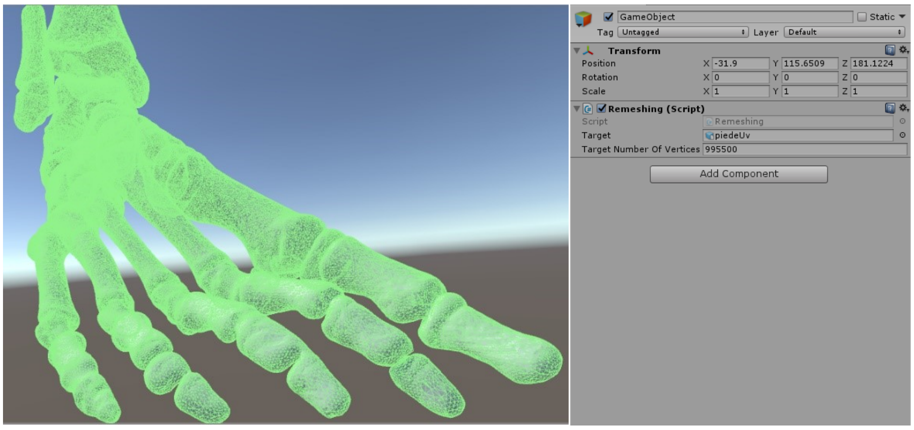
From the conducted tests, the optimal set of parameters resulted a total number of polygonal faces less than 4500 and a total number of vertices approximately of 10000 vertices. With these parameters, the cutting time has been reduced and the waiting time from the user input to the actual cut of the bone resulted in approximately 10 seconds [Table 7].



3D model	Vertices	Faces	Cutting time [s]
Exostosis	11634	4312	9
Hip bone	10586	4489	10
Child femur	4470	4112	5
Tibia	6443	4269	6
Foot	12329	4400	11
Adult femur	7436	4201	6

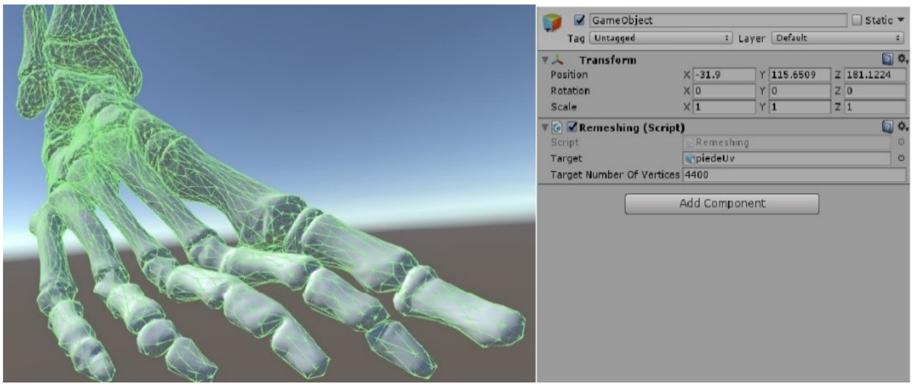
Table 7

In order to reduce mesh complexity and cut any imported bone without losing the physical resolution and appearance of the bone, we implemented a C# script for Unity runtime mesh decimation and simplification. The decimation script is based on the Ramer – Douglas – Peucker algorithm and it allows to reduce the number of points of a line, while maintaining the line shape. Moreover, the script allows the user to set the final maximum number of vertices that the cut object would have. In fig. [Fig. 5.26] and in fig. [Fig. 5.27] it is depicted the application of the decimation algorithm to the foot 3D model.



☑ GameObject □ Static
Tag: Untagged Layer: Default
Transform
Position X -31.9 Y 115.6509 Z 181.1224
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
☑ Remeshing (Script)
Script: Remeshing
Target: piedeUv
Target Number Of Vertices: 995500

FIGURE 5.26: 3D model before decimation



☑ GameObject ☐ Static
Tag Untagged Layer Default

Transform	X	Y	Z
Position	-31.9	115.6509	181.1224
Rotation	0	0	0
Scale	1	1	1

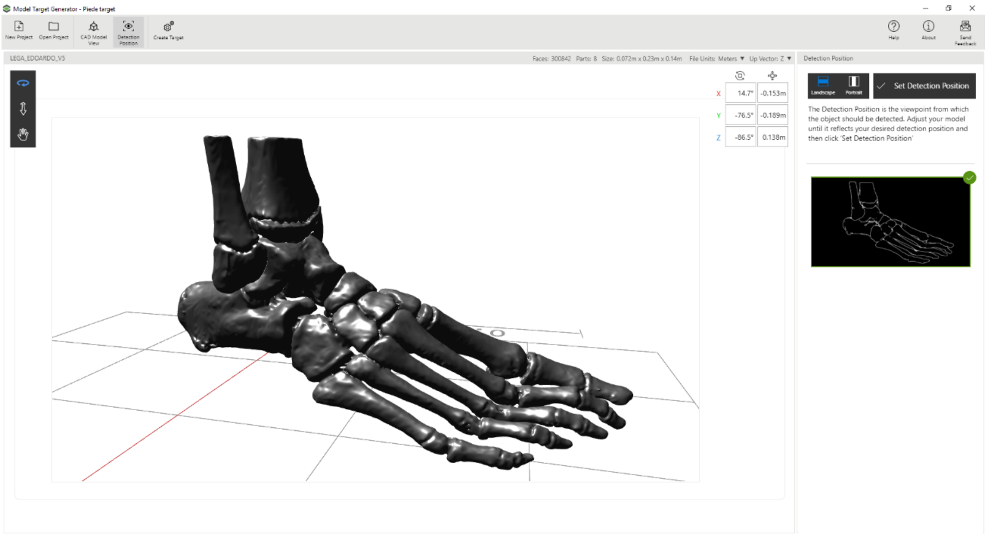
☑ Remeshing (Script)
Script Remeshing
Target piedeUv
Target Number Of Vertices 4400

FIGURE 5.27: 3D model after decimation

The Ramer – Douglas – Peucker algorithm is particularly suitable for our application because it is fast enough to be implemented in runtime.

5.3.5 Mixed reality set – up

Once the cutting script has been developed, we developed the mixed reality environment. The immersive environment has been developed in Unity3D and it leverages the Vuforia SDK as augmentation library. In particular, the application implements the innovative Vuforia CAD model target component, which is an upgrade of the Vuforia object target described in Chapter XXX. Vuforia CAD model target allows to use the polygonal 3D model to create a visual target which is used as target to augment the real scene. This is a step forward with respect to the previous Vuforia object target since the real object does not require any more to be scanned and the 3D model, derived from a CAD parametric model or from other sources, can be used to create a fiducial marker in a complete digital workflow [Fig. 5.28].



LEGA_EDOARDO_V5

Faces: 300842 Parts: 8 Size: 0.072m x 0.23m x 0.14m File Units: Meters Up Vector: Z

X	14.7°	-0.153m
Y	-76.5°	-0.189m
Z	-86.5°	0.138m

Detection Position

The Detection Position is the viewpoint from which the object should be detected. Adjust your model until it reflects your desired detection position and then click 'Set Detection Position'

FIGURE 5.28: Creation of the target from the 3D model in the Vuforia CAD model target application

Then, the generated target can be imported in the Unity project and use as database of points for the tracking. The use of Vuforia CAD model target presents some drawbacks. First of all, to begin tracking of the real object and accurately superimposing the virtual content it is mandatory to frame the real object in the same exact position in which the model has been framed in the Vuforia CAD model target generator. This means that the choice of the initial framing position is crucial and it affects the entire user experience. The second drawback of the use of Vuforia CAD model target depends on the color of the 3D printed model. In medical field, bony models have always been made in white material but in our case 3D printed models in white filament generate low quality tracking. The bad tracking is caused by the reflections of white material under room lights. For this reason, the 3D printed model has been printed in black matte material in order to reduce reflections and to enhance tracking capabilities [Fig. 5.29].

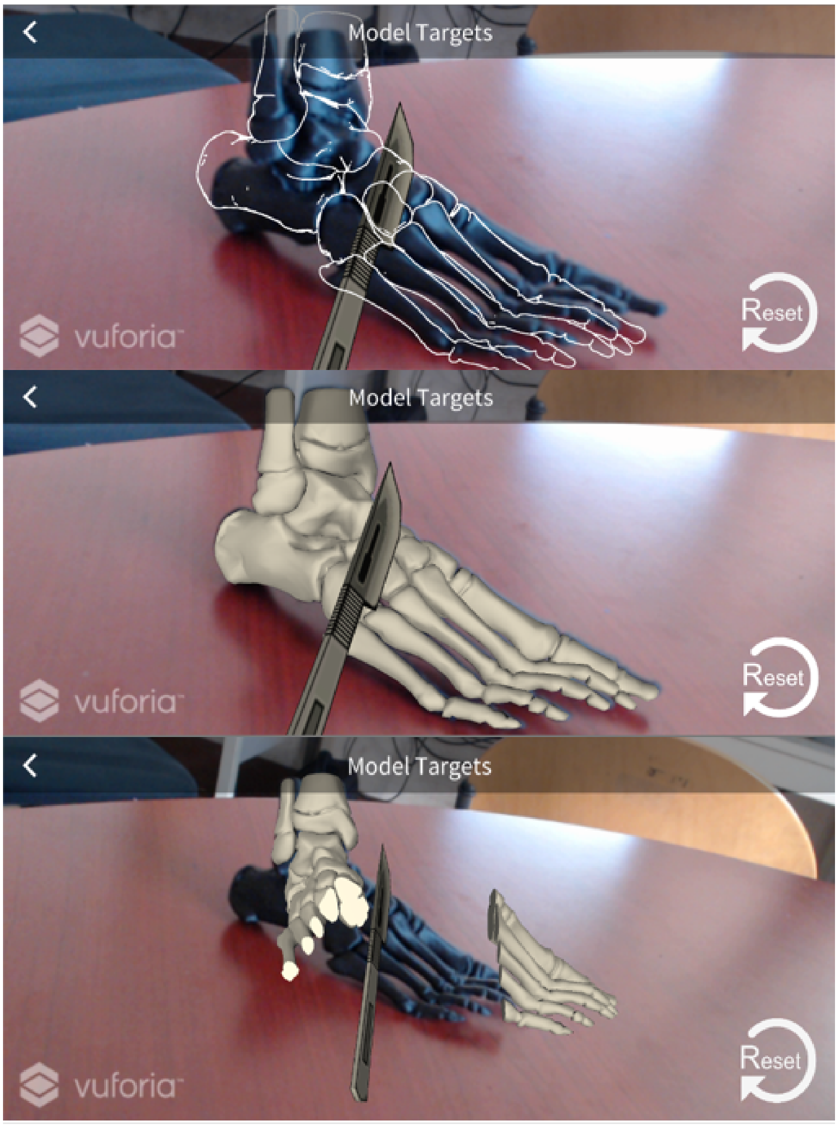


FIGURE 5.29: Tracking of the 3D printed model with Vuforia CAD model target

HoloLens V1 supports only single finger recognition. This kind of interaction in the surgical field is quite limiting. Furthermore, while the gesture has been conceived by Microsoft developers to be simple in theory, it has been demonstrated that the picking gesture is quite difficult for novice or new users to perform. The encountered difficulties in the interaction can be annoying and frustrating for the user and, in the end, this determines the success of an immersive technology application and whether or not the development of the application would continue. To overcome these limitations, we decided to implements LeapMotion controller to enables the tracking of the entire hands. To test the feasibility of the LeapMotion implementation, a prototype of the application has been developed and the hardware set – up comprehends a webcam and LeapMotion controller [Fig. 5.30].

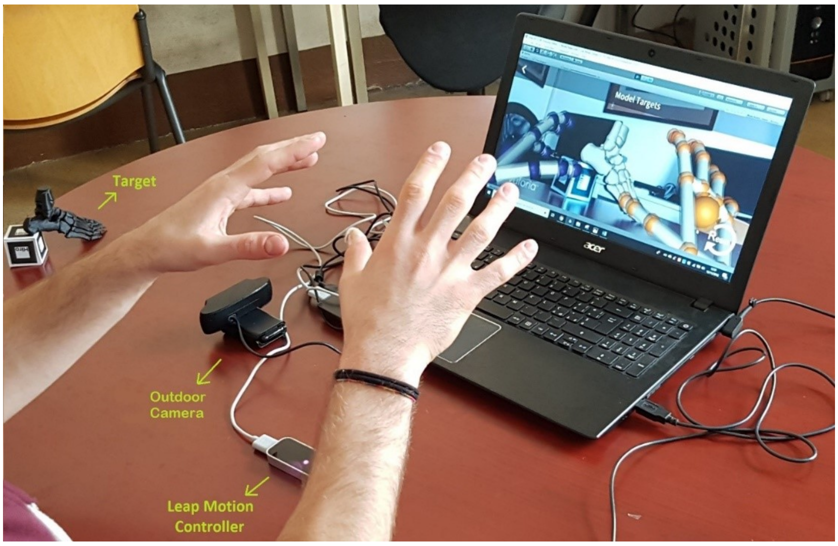


FIGURE 5.30: Prototype of the application in a desktop set - up

The second version of the application has been developed for the integration of LeapMotion controller and Microsoft HoloLens. The LeapMotion controller has been mounted on the upper frame of HoloLens with a certain angle of inclination in order to have the depth sensor of LeapMotion at the optimum angle and to frame the hands of the user [Fig. 5.31].

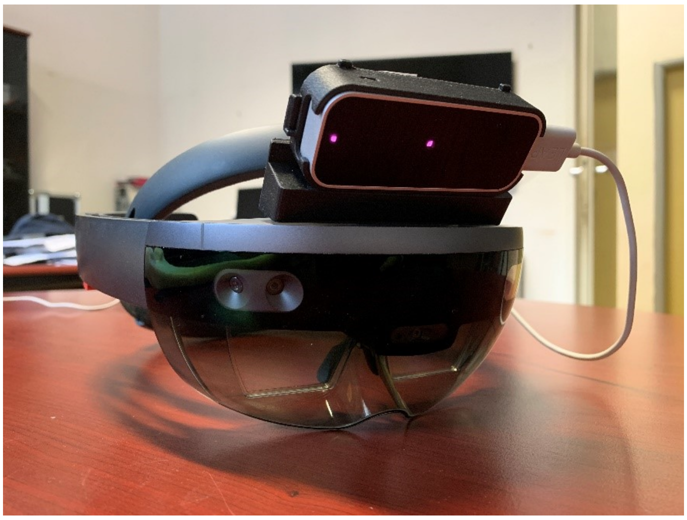


FIGURE 5.31: Integration of LeapMotion in HoloLens

LeapMotion controller could not be directly connected to Microsoft HoloLens since HoloLens does not support the use of peripherals. Hence, LeapMotion has been connected to a laptop and a UDP connection between HoloLens and the laptop has been established in order to stream data from the LeapMotion to HoloLens. In [Fig. 5.32] the user is handling a femur and in the screen in the background the tracking of the user hand by LeapMotion and the tracking of the bone model by Vuforia CAD model target are performed.

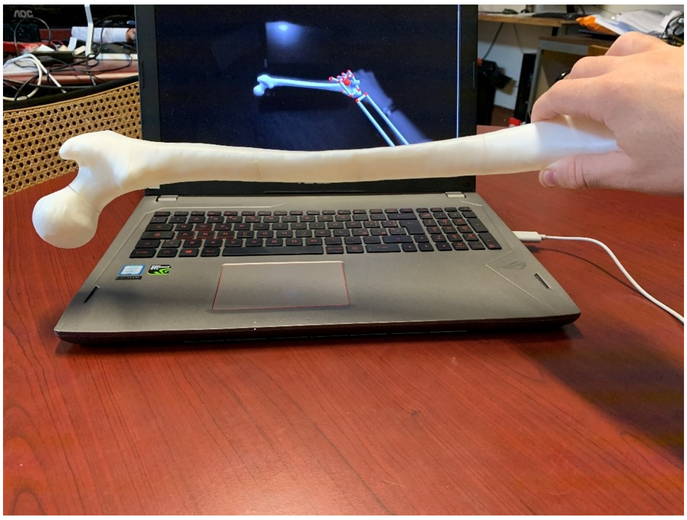


FIGURE 5.32: Simultaneous tracking of the user hand by LeapMotion and tracking of the bone by Vuforia

The UDP connection minimizes the latency and allows a high speed flow of data. As mentioned before, the drawback of the use of UDP connection is that the data would not be delivered to the client, or that the data are delivered in an incorrect order. However, the data sent form LeapMotion controller are high frequency data, approximately 100 frame per second, and the data are dispatched with an incrementing ID which make the errors less frequent to happen. In the sequence of figures [Fig. 5.33, 5.34, 5.35] it is depicted the cutting operation in the first person view from the user perspective, wearing HoloLens. In [Fig. 5.33] the user is using the scalpel, to which has been attached a visual indicators of the cutting plane, to pose the cutting line.

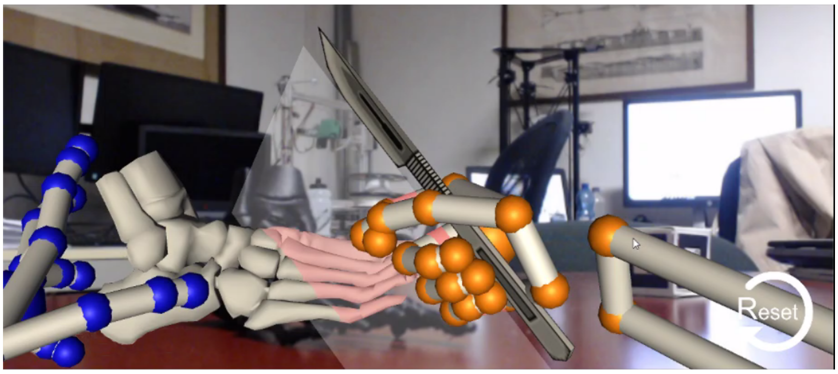


FIGURE 5.33: The user is placing the cutting plane

In [Fig. 5.34] the input of the user, in this case the left hand closing gesture, operates the cut.

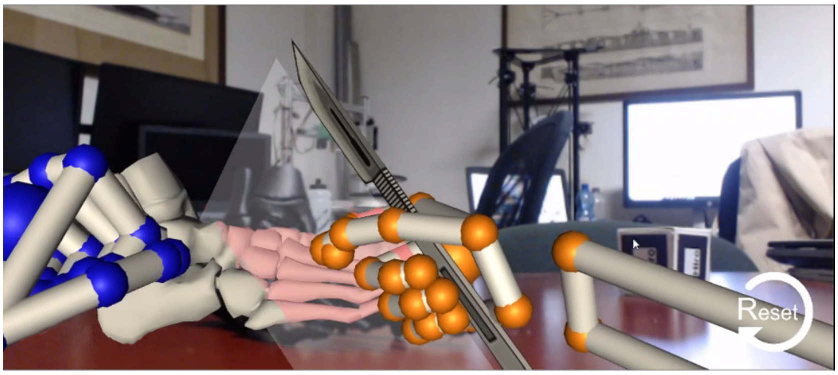


FIGURE 5.34: The user is giving the cutting input by closing the left hand

In [Fig. 5.35] the user has released the scalpel and can manipulate the two halves of the foot.

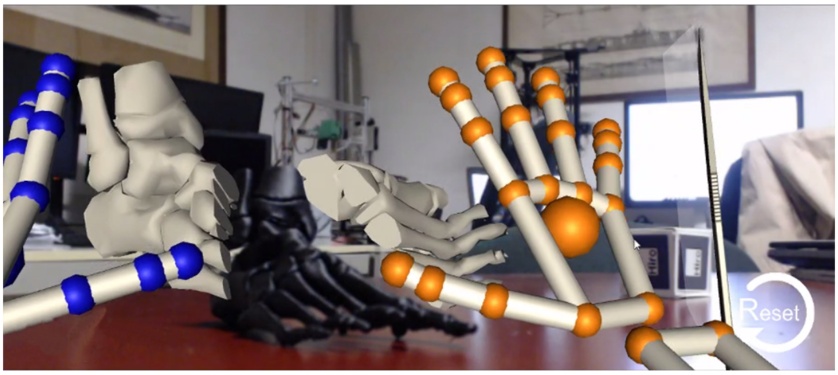


FIGURE 5.35: The user interacts with the two halves

5.3.6 Results

In this research we have proven the feasibility of the use of immersive technologies in medical field. In particular, the project successfully developed a workflow to convert DICOM data of patient CT scans into 3D models that can be used for pre – operative planning. The 3D printed model of the patient bone can be augmented with virtual content and the surgeon can simulate multiple surgical strategies. The mixed reality set – up has involved the development of a bridge connection between Microsoft HoloLens and LeapMotion controller in order to add more intuitive gestures to the already built – in HoloLens gestures. The connection has been set – up via UDP in order to maintain the data exchange as fast as possible to guarantee real – time response of the application. The real – time response of the application also depends on the resolution of 3D model of the bone. For this reason, a specific C# script has been developed in order to integrate the Ramer – Douglas – Peucker decimation algorithm in the runtime of the application. This allowed to guarantee a response time on user input of approximately 10 seconds. The result of the use of this application is that the surgeon can study the disease wearing the AR HMD, test and evaluate several surgical strategies and then validate the surgical plan on the 3D printed model [Fig. 5.36].

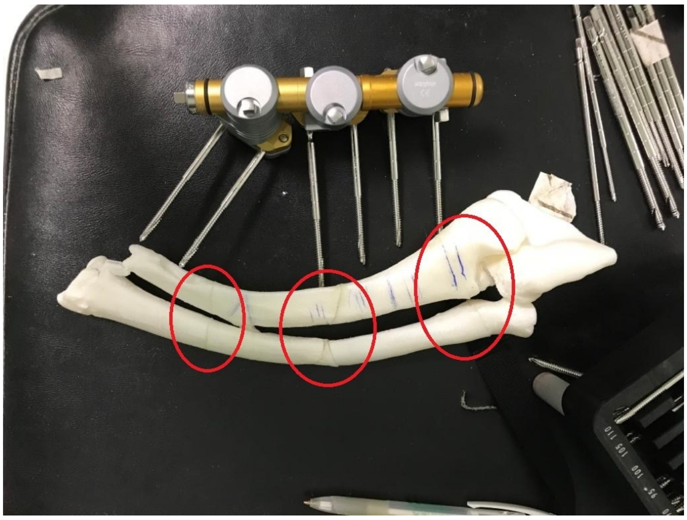
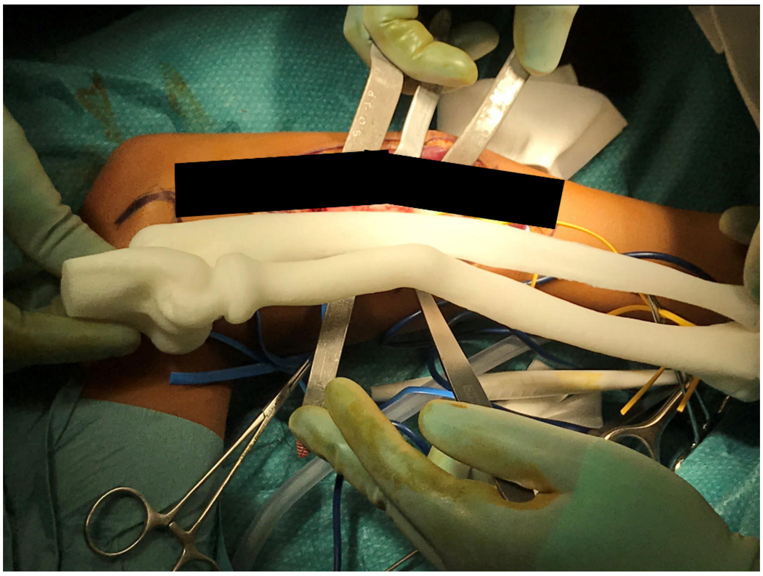


FIGURE 5.36: Cutting test on the real 3D printed model

5.3.7 Conclusions

Besides the pre – operative planning, the use of immersive technologies can be exploited to improve training of young surgeon or provide remote support to surgeons both in the pre – operative planning and during the surgical operations. Furthermore, the use of 3D models, either printed or virtual, can enhance the surgeon diagnosis of the disease and improve the decision making process in the pre – operative phase: most of the decisions in this phase are taken only by using CT scans on 2D hardware, like pc screens, or X ray images. Hence, not making the most of 3D reconstruction technology provided by CT scanners. On the other hand, it could happen that in the operative room the patient may undergo further tests and investigation by means of intra-operative x-rays to evaluate and decide on – the – fly the best surgical procedure to follow. To overcome the use of intraoperative X – rays, this research was part o f a wider project aiming to introduce industry 4.0 tool in the medical field. Besides the use of AR/VR in pre – operative planning, the project developed a workflow for 3D printing bony models from CT scans to use in the operating rooms. The workflow allows quick conversion of CT scans into 3D printable models and the 3D models are printed in High Temperature PLA (HTPLA). This material after an annealing process, can withstand the temperature of low pressure sterilization process used by IOR. Thanks to this workflow, 3D printed bony model of the patient can be used in operating rooms successfully in place of unnecessary intraoperative x – rays [Fig. 5.37].



[redacted]

FIGURE 5.37: Use of the 3D printed model of the patient in the operating room

Chapter 6

Conclusions

This research developed a new tool for the early stage design, based on hybrid modeling techniques, capable of merging the quick edits to the geometry, operated in the immersive environment, with the procedural high - level rules for assembly features definition and coupling components. In particular, the new tool supports designers in the design of assembly and the evaluation of the feasibility of the assembly and disassembly sequence. The design of assembly and disassembly sequence leverages the interactive capabilities of the immersive environment; thus, enriching with fruitful information the decision making process and, in the end, boosting up the design process. The designer no longer relies only on intangible tools and with little connection to reality, like spreadsheets or optimization algorithms, but he can take advantage of the near - tangible access to the product and evaluate the design choices by a hand - on approach. At the moment, the design and optimization of assembly and disassembly are promising field of application of immersive technologies. In these fields, virtual reality and augmented reality can be effectively exploited. The research implemented AR and VR in assembly and disassembly operations for three main tasks:

1. Design and virtual prototyping of assembly features in the immersive environment with a data exchange workflow between the Unity3D game engine and the high – level set of rules in the Grasshopper modeling environment;
2. Evaluation of the design of joints and assessment of the disassembly sequence through a hands – on approach in an augmented reality environment;
3. Evaluation of the capabilities of immersive technologies for new workforce training and manual skills development.

As a result of the research described in this thesis, we can highlight the following conclusions:

1. The data exchange flow between Unity and Grasshopper is able to provide a robust modeling system of joints and couplings between different components directly in the immersive environment. The modeling system in Unity3D is based on a polygonal representation but the input is converted into Grasshopper input. Once in Grasshopper the input can be used to create B – rep geometries or mesh geometries, accordingly to the Grasshopper blocks used;
2. A library of basic assembly relationships has been implemented in Grasshopper's Kangaroo, including: face to face constraint; hinge constraint; pin/hole constraint and slider constraint;

3. The assembly constraints are set using the input of the immersive device, in our case the HTC Vive controller, and then used to define the geometric elements necessary as input of the Kangaroo blocks in the Grasshopper environment. The collision detection operated by Unity3D engine allows the user to freely set the assembly relationships, avoiding the overlapping of the models involved in the assembly operations;

4. Designers and workers can test the disassembly sequence right after designing it in order to speed up the designing process, preventing design issues. Moreover, the disassembly sequence can be modified accordingly to the suggestions of the workers involved in the design process. These suggestions can leverage the experience that long time workers had gained in manufacturing, service and maintenance operations. This will allow to conceive the most effective and the most suitable assembly and disassembly sequence for new workers. Lastly, this allow to convert the experience of workers into a form of digital data that will enrich and consolidate the company know – how;

5. The immersive environment can then be used to train workers in the assembly and disassembly sequence. The use of VR and AR for training purposes has been studied and the advantages of this type of training have been described and compared to the traditional training method, for example decreased training times. Moreover, training in VR can create a deeper involvement in workers, increasing the learning rate of workers.

6. Besides the research on the application of immersive technologies in industrial scenarios, we also focused our research in the use of immersive technologies in other fields. While the use of immersive technologies in industrial context is manly aimed to transfer information, such as the correct step of the assembly sequence; in other field where more manual interaction is required, immersive technologies can improve and accelerate the development of manual skills capabilities of the worker. In such a perspective, the work conducted at the Oregon State University for AEC sector was intended for this purpose.

7. In medical field, AR and VR allow students to comprehend the medical subjects through a hand – on approach. The learning process is transformed: the simple memorization of concepts is integrated to first person application, analysis and evaluation of the diseases extracted from real – life cases. AR and VR medical applications can raise the level of engagement of students and deliver a faster learning improved by critical thinking and problem-solving capabilities.

6.1 Future research

The work described in this thesis highlights that the use of immersive technologies will increase both in numbers and in application sectors. Consequently, the research in this field will grow. For this reason, some improvements in the research field in immersive technologies should be taken into account.

First of all, the diversification of the immersive applications leads to a diversification of sample used to validate prototype or case studies. In most research works [57], even in this work, researchers tend to use students as participants of the experiments. The use of student can be good for a rough evaluation of the application, like usability, UX etc., but, in the end, the application has to be tested with final end users in order to evaluate its real effectiveness.

One of the most common and well – known drawback of the use of AR and VR is cybersickness. Cybersickness can appear as simple motion sickness and eye sickness as well as nausea. In some users, the symptoms appear after few minutes of use, while in others after long immersive sessions. Another research trend could be the identification and the conceptualization of solutions to cybersickness problem. This future research trend could involve also UX research since cybersickness and discomfort are strictly related to the quality of the experience perceived by the user.

AR and VR are changing the way we interact with world; hence a development of new hardware is necessary in order to render these technologies as portable as possible. At the moment, only Microsoft HoloLens has internal computational capabilities. All the other HMDs available on the market require to be connected to a high level desktop. Reducing the minimum hardware capabilities and making the hardware more portable will improve the diffusion of these technologies. The portability and the freedom of movements represent the main advantage of the development of more powerful self-contained HMD. For HMD that requires external computational power, the backpack set – up available with few high end computers is not suitable for industrial applications for two reasons. Firstly, the backpack hardware set – up is pretty expensive and more important, the backpack hardware is extremely bulky and unwieldy to be used in an industrial scenario with confidence. Lastly, with HTC Vive and other HMD that implements external tracking system, the backpack set – up is not suitable at all.

Another limitation of current hardware for immersive technologies is related to visibility. In terms of Field of View (FoV), HMD current on the market has a narrow FoV. The largest FoV current available on the market is provided by Meta2 with its 90 degree FoV, which is 30% less than human eye FoV. All HMD, both VR or AR, need wider FoV to enhance UX. Specific upgrades need to be carried out for AR HMD which require an improvement of visibility in different lighting conditions. For example, it is almost impossible to see augmented virtual content with HoloLens in full sun light.

Authoring virtual content is extremely time consuming. Despite Unity3D has a learning curve pretty high, a development of a new generation ad more agile software of scene editor should be carried out. Regarding new authoring software, high end CAD provider, like PTC, has started to develop a new suite of software tool, in between CAD modeling and content authoring. Vuforia Studio is intended to be an easier, but less powerful compared to Unity3D, scene editor strictly connected to PTC Creo Parametric. On the other hand, Vuforia Chalk is off the shelf solution for rapid remote guidance and remote assistance. However, PTC is the only CAD provider that has seen the potential of immersive technologies and has acquired immersive technologies developers. Other software houses rely on plug – in by external developers.

Immersive technologies allow users to interact in an intuitive way with digital data and content, reducing the distance between people and information and delivering fully immersive experience. With the use of immersive headset workers can connect with colleagues to find solutions to problems in a faster and more seamless way. The information is displayed in the HMD to users as they need it, enriching the understanding of design and manufacturing issues and improving the decision making process.

Chapter 7

Bibliography

[1] PTC surveys “CAD and Complexity”, 300+ respondents and “Concept Design Challenges”,200+ respondents, 2011.

[2] J. Yang, S. Han, H. Kang, and J. Kim, “Product data quality assurance for e-manufacturing in the automotive industry,” International Journal of Computer Integrated Manufacturing, vol. 19, no. 2, pp. 136–147, 2006.

[3] H. Gu, T. R. Chase, D. C. Cheney, T. “Tim” Bailey, and D. Johnson, “Identifying, Correcting, and Avoiding Errors in Computer-Aided Design Models Which Affect Interoperability,” Journal of Computing and Information Science in Engineering, vol. 1, no. 2, pp. 156–166, May 2001.

[4] ISO 10303-43. Industrial automation systems and integration—Product data representation and exchange. Integrated generic resource: Representation structures. Geneva (Switzerland): International Organization for Standardization; 2011.

[5] C. González-Lluch, P. Company, M. Contero, J. D. Camba, and R. Plumed, “A survey on 3D CAD model quality assurance and testing tools,” Computer-Aided Design, vol. 83, pp. 64–79, 2017.

[6] ISO 10303-108. Industrial automation systems and integration—Product data representation and exchange: Integrated application resource: Parameterization and constraints for explicit geometric product models. Geneva (Switzerland): International Organization for Standardization; 2005.

[7] I. C. Braid, “The Synthesis of Solids Bounded by Many Faces,” Commun. ACM, vol. 18, no. 4, pp. 209–216, 1975.

[8] G. P. Qian and R. F. Tong, “Automatical Reconstruction of Deficient CAD Model,” Advanced Materials Research, vol. 186, pp. 241–245, Jan. 2011.

[9] Gilles Foucault and Jean-Christophe Cuilliere and Vincent Francois and Jean-Claude Leon and Roland Maranzana, “Adaptation of CAD model topology for finite element analysis,” Computer-Aided Design, vol. 40, no. 2, pp. 176–196, Feb. 2008.

[10] S. Bischoff and L. Kobbelt, “Structure Preserving CAD Model Repair,” Computer Graphics Forum, vol. 24, no. 3, pp. 527–536, Sep. 2005.

[11] A. R. Grayer, “Alternative approaches in geometric modelling,” Computer-Aided Design, vol. 12, no. 4, pp. 189–192, Jul. 1980.

[12] Y. Woo and S. H. Lee, "Volumetric modification of solid CAD models independent of design features," Advances in Engineering Software, vol. 37, no. 12, pp. 826–835, Dec. 2006.

[13] B. C. Kim and D. Mun, "Stepwise volume decomposition for the modification of B-rep models," The International Journal of Advanced Manufacturing Technology, vol. 75, no. 9–12, pp. 1393–1403, Aug. 2014.

[14] J. Fu, X. Chen, and S. Gao, "Automatic synchronization of a feature model with direct editing based on cellular model," Computer-Aided Design and Applications, vol. 14, no. 5, pp. 680–692, Jan. 2017.

[15] M. Lipp, P. Wonka, and P. Müller, "PushPull++," ACM Transactions on Graphics, vol. 33, no. 4, pp. 1–9, Jul. 2014.

[16] F. Bernardini, C. L. Bajaj, J. Chen, and D. R. Schikore, "automatic reconstruction of 3d cad models from digital scans," International Journal of Computational Geometry & Applications, vol. 9, no. 04n05, pp. 327–369, Aug. 1999.

[17] J. J. Shah and M. T. Rogers, "Assembly modeling as an extension of feature-based design," Research in Engineering Design, vol. 5, no. 3, pp. 218–237, Sep. 1993.

[18] D. E. Whitney, R. Mantripragada, J. D. Adams, and S. J. Rhee, "Designing assemblies," Research in Engineering Design - Theory, Applications, and Concurrent Engineering, vol. 11, no. 4, pp. 229–253, 1999.

[19] W. van Holland and W. F. Bronsvoort, "Assembly features in modeling and planning," Robotics and Computer-Integrated Manufacturing, vol. 16, no. 4, pp. 277–294, Aug. 2000.

[20] C. Lu, J. Y. H. Fuh, Y. S. Wong, W. D. Li, and Y. Q. Lu, "Feature-Based Design Modification in Co-Assembly Design," Computer-Aided Design and Applications, vol. 1, no. 1–4, pp. 411–420, Jan. 2004.

[21] S. Rachuri et al., "A Model for Capturing Product Assembly Information," Journal of Computing and Information Science in Engineering, vol. 6, no. 1, pp. 11–21, Jun. 2005.

[22] S. Jayaram, U. Jayaram, Y. Wang, H. Tirumali, K. Lyons, and P. Hart, "VADE: a Virtual Assembly Design Environment," IEEE Computer Graphics and Applications, vol. 19, no. 6, pp. 44–50, 1999.

[23] Y.-S. Ma, G. A. Britton, S. B. Tor, and L. Y. Jin, "Associative assembly design features: concept, implementation and application," The International Journal of Advanced Manufacturing Technology, vol. 32, no. 5–6, pp. 434–444, May 2006.

[24] S. M. M. Murshed, J. J. Shah, V. Jagasivamani, A. Wasfy, and D. W. Hislop, "OAMmathplus: An Assembly Data Model for Legacy Systems Engineering," in Volume 6: 33rd Design Automation Conference, Parts A and B, 2007.

[25] W. Tao, Y. Qing-dong, and L. Hong-cai, "Research on method of virtual modeling based on reconstruction of virtual assembly feature," in 2010 International Conference on Audio, Language and Image Processing, 2010.

[26] J. Zhang, Z. Xu, Y. Li, and S. Jiang, "Framework for the integration of assembly modeling and simulation based on assembly feature pair," The International Journal of Advanced Manufacturing Technology, vol. 78, no. 5, pp. 765–780, May 2015.

[27] M. Fa, T. Fernando, and P. M. Dew, "Interactive constraint-based solid modeling using allowable motion," in Proceedings on the second ACM symposium on Solid modeling and applications - SMA & 93, 1993.

[28] J. R. Rossignac, "Constraints in constructive solid geometry," in Proceedings of the 1986 workshop on Interactive 3D graphics - SI3D & 86, 1987.

[29] A. Stork and M. Maidhof, "Efficient and precise solid modelling using a 3D input device," in Proceedings of the fourth ACM symposium on Solid modeling and applications - SMA & 97, 1997.

[30] S. Gao, H. Wan, and Q. Peng, "An approach to solid modeling in a semi-immersive virtual environment," Computers & Graphics, vol. 24, no. 2, pp. 191–202, Apr. 2000.

[31] W. Ma, Y. Zhong, S.-K. Tso, and T. Zhou, "A hierarchically structured and constraint-based data model for intuitive and precise solid modeling in a virtual reality environment," Computer-Aided Design, vol. 36, no. 10, pp. 903–928, Sep. 2004.

[32] Z. Liu and J. Tan, "Constrained behavior manipulation for interactive assembly in a virtual environment," The International Journal of Advanced Manufacturing Technology, vol. 32, no. 7, pp. 797–810, Apr. 2007.

[33] J. J. Roldan, "A training system for Industry 4.0 operators in complex assemblies based on virtual reality and process mining," Robotics and Computer-Integrated Manufacturing, vol. 59, pp. 305–316, Oct. 2019.

[34] N. Ho, P.-M. Wong, M. Chua, and C.-K. Chui, "Virtual reality training for assembly of hybrid medical devices," Multimedia Tools and Applications, vol. 77, no. 23, pp. 30651–30682, Dec. 2018.

[35] B. Nash, A. Walker, and T. Chambers, "A simulator based on virtual reality to dismantle a research reactor assembly using master-slave manipulators," Annals of Nuclear Energy, vol. 120, pp. 1–7, Oct. 2018.

[36] R. De Amicis, A. Ceruti, D. Francia, L. Frizziero, and B. Simoes, "Augmented Reality for virtual user manual," International Journal on Interactive Design and Manufacturing (IJIDeM), vol. 12, no. 2, pp. 689–697, May 2018.

[37] J. Chalhoub and S. K. Ayer, "Using Mixed Reality for electrical construction design communication," Automation in Construction, vol. 86, pp. 1–10, Feb. 2018.

[38] C.-J. Chao, S.-Y. Wu, Y.-J. Yau, W.-Y. Feng, and F.-Y. Tseng, "Effects of three-dimensional virtual reality and traditional training methods on mental workload and training performance," Human Factors and Ergonomics in Manufacturing & Service Industries, vol. 27, no. 4, pp. 187–196, May 2017.

[39] O. Cakmakci and J. Rolland, "Head-Worn Displays: A Review," J. Display Technol., vol. 2, no. 3, pp. 199–216, Sep. 2006.

[40] K. Otao, Y. Itoh, K. Takazawa, H. Osone, and Y. Ochiai, "Air Mounted Eyepiece," in Proceedings of the 9th Augmented Human International Conference on - AH & 18, 2018.

[41] Meta Official Website. Available from: https://meta-eu.myshopify.com/, visited on 02/28/2018.

[42] N. Beattie, B. Horan, and S. McKenzie, "Taking the LEAP with the Oculus HMD and CAD - Plucking at thin Air?" Procedia Technology, vol. 20, pp. 149–154, 2015

[43] Alvar VTT website, http://virtual.vtt.fi/virtual/proj2/multimedia/alvar/index.html, visited on 01/27/2017.

[44] W. Pan, Y. Wang, and P. Du, "Automatic disassembly navigation for accurate virtual assembly path planning," Assembly Automation, vol. 34, no. 3, pp. 244–254, Jul. 2014.

[45] MIL-STD-1472G, Department of Defense Design Criteria Standard: Human Engineering (11-JAN-2012).

[46] U.S. Department of Labor/Bureau of Labor Statistics. (2018). Employment Projections and Occupational Outlook Handbook. Washington, DC.

[47] McGraw-Hill Construction, Construction industry workforce shortages: role of certifications, training and green jobs in filling the gaps. Bedford, MA: McGraw-Hill Construction, 2012.

[48] D. L. Kirkpatrick, "Seven keys to unlock the four levels of evaluation," Performance Improvement, vol. 45, no. 7, pp. 5–8, 2006.

[49] J. Chalhoub and S. K. Ayer, "Using Mixed Reality for electrical construction design communication," Automation in Construction, vol. 86, pp. 1–10, 2018.

[50] F. D. Davis, "Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology," MIS Quarterly, vol. 13, no. 3, pp. 319–340, 1989.

[51] J. J. Roldán, E. Crespo, A. Martín-Barrio, E. Peña-Tapia, and A. Barrientos, "A training system for Industry 4.0 operators in complex assemblies based on virtual reality and process mining," Robotics and Computer-Integrated Manufacturing, vol. 59, pp. 305–316, 2019.

[52] N. Ho, P.-M. Wong, M. Chua, and C.-K. Chui, "Virtual reality training for assembly of hybrid medical devices," Multimedia Tools and Applications, vol. 77, no. 23, pp. 30651–30682, Dec. 2018.

[53] Anderson, L.W. (Ed.), Krathwohl, D.R. (Ed.), Airasian, P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R., Raths, J., & Wittrock, M.C. (2001). A taxonomy for learning, teaching, and assessing: A revision of Bloom's Taxonomy of Educational Objectives (Complete edition). New York: Longman.

[54] C. Sagnier, E. Loup-Escande, and Geacute, "Effects of Gender and Prior Experience in Immersive User Experience with Virtual Reality," in Advances in Intelligent Systems and Computing, Springer International Publishing, 2019, pp. 305–314.

[55] U.S Department of Labor, Women's Bureau https://www.dol.gov/wb/occupations_interactive.html accessed on 19/10/2019.

[56] S. Hoedt, A. Claeys, H. V. Landeghem, and J. Cottyn, "The evaluation of an elementary virtual training system for manual assembly," International Journal of Production Research, vol. 55, no. 24, pp. 7496–7508, Sep. 2017.

[57] A. Suh and J. Prophet, "The state of immersive technology research: A literature analysis," Computers in Human Behavior, vol. 86, pp. 77–90, Sep. 2018.

Appendix A

Preliminary test

Ten questions for the pre – training test are selected from the list below. Questions are divided for each task. The questionnaire has two questions for each task.

A.1 Task 1: Recognize the material

1. Can you list the names of all studs used in the wall framing process? If Yes:
2. Can you describe the difference between king studs and jack studs? If Yes:
3. Can you draw the layout of a wall with a framed door? If Yes:

A.2 Task 2: Wall layout

1. Can you write the "16 on center" rule? If yes:
2. Can you say how king studs, jack studs and cripple studs are marked on the top/bottom plate during layout? If Yes:
3. Can you say where you retrieve the length of the door rough opening? If Yes:
4. Can you say which component has the same length of the door rough opening? If Yes:
5. With respect to the length of sheathing panels, can you explain why the "16 on center" rule is implemented? If Yes:
6. Can you say why you mark both sides of the studs onto the top and bottom plate? If Yes:

A.3 Task 3: Wall framing

1. Can you say what toe nailing is? If Yes:
2. Can you say when toe nailing is applied? If Yes:
3. Can you say what distance from the stud edge you should nail to avoid the stud splitting? If Yes:
4. Can you describe one method to prevent wavy sheathing? If Yes:
5. With respect to the execution procedure and the goals, can you describe the difference between nailing and toe nailing? If Yes:

6. Can you say why the studs are laid down with crowns in the same direction? If Yes:
7. Can you say why toe nailing is applied? If Yes:

A.4 Task 4: Wall sheathing

1. Can you list best practices to properly align sheathing? If Yes:
2. Can you define what the shear strength is? If Yes:
3. Can you say the first thing to do when adjusting the blade depth of a circular saw? If Yes:
4. Can You say what it is used as a reference to align the sheathing panel with the studs? If Yes:
5. Can you say why using the edges of factory sheathing to align sheathing panels? If Yes:
6. Can you say how to maximize shear strength using sheathing panels? If Yes:
7. Can you say why the blade depth is set equal to the panel thickness? If Yes:

A.5 Task 5: Wall standing

1. Can you list safety precautions for a wall standing procedure? If Yes:
2. Can you say how many workers are required to stand a 6 feet wall? If Yes:
3. Can you say what it can happen when lifting a wall? If Yes:
4. Can you say why it is necessary to use a long level instead of a short one? If Yes:
5. Can you say what the workers should do, if the wall starts falling? If Yes:
6. Can you say where you should nail the temporary braces on the floor? If Yes:
7. Can you say why the wall is lifted with a gap between the bottom plate and the edge of the floor system? If Yes:
8. Can you say why the temporary brace should be nailed to the higher part of the wall studs? If Yes: