

Doctoral Dissertations and Master's Theses

---

Fall 11-2017

## Computer Aided Drafting Virtual Reality Interface

David M. Roach  
*Embry-Riddle Aeronautical University*

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Computer Engineering Commons](#)

---

### Scholarly Commons Citation

Roach, David M., "Computer Aided Drafting Virtual Reality Interface" (2017). *Doctoral Dissertations and Master's Theses*. 374.

<https://commons.erau.edu/edt/374>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Doctoral Dissertations and Master's Theses by an authorized administrator of Scholarly Commons. For more information, please contact [commons@erau.edu](mailto:commons@erau.edu).

# Computer Aided Drafting Virtual Reality Interface

by

David M. Roach

A Thesis Submitted to the College of Engineering Department of Electrical,  
Computer, Software, & Systems Engineering in Partial Fulfillment of the  
Requirements for the Degree of Master of Science in Computer Engineering

November 2017

Embry Riddle Aeronautical University

Daytona Beach, Florida

Computer Aided Drafting Virtual Reality Interface

by

David M. Roach

This thesis was prepared under the direction of the candidate's thesis committee Chair Dr. Ilteris Demirkiran, Department of Electrical, Computer, Software & Systems Engineering, and has been approved by the members of the thesis committee. It was submitted to the Department of Electrical, Computer, Software & Systems Engineering and was accepted in partial fulfillment of the requirements for the degree of Master of Science in Computer Engineering.

THESIS COMMITTEE



Dr. Ilteris Demirkiran, Committee Chair



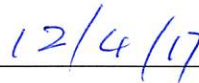
Dr. Shuo Pang, Committee Member



Dr. Tianyu Yang, Committee Member,



Dr. Timothy A Wilson, Department Chair  
or Dr. Jianhua Liu, Graduate Program Chair



Date



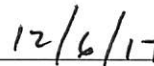
Dr. Maj Mirmirani, Dean of College of Engineering



Date



Dr. Christopher D. Grant, Vice Chancellor for  
Academics



Date

## ACKNOWLEDGEMENTS

I would like to extend a heartfelt expression of gratitude to the esteemed professors on my graduate thesis committee, Dr. Shuo Pang, Dr. Tianyu Yang, and my graduate thesis advisor, Dr. Ilteris Demirkiran, whose guidance and mentorship helped mold and direct this project. Please know that your willingness to work with me as part of my thesis committee and the feedback you offer are invaluable to me and I appreciate it immensely. Also, thank you to my two delightful children, Matthew and Ryan, who deeply enjoyed their first foray into the wonderful world of virtual reality during their summer break, and their mother, Misty, who gracefully endured and accepted the need for long hours spent on this meaningful endeavor. Thank you to my parents, who continue to inspire me to work harder and smarter in all aspects of my life. And thank you to my brothers, Tyler and Wesley, who motivate me to fully pursue higher education in order to reach my full potential.

## TABLE OF CONTENTS

I.	Abstract .....	7
II.	Introduction .....	8
III.	Literature Review .....	8
IV.	Preliminary Research .....	31
V.	Method.....	41
VI.	Results .....	61
VII.	Conclusion.....	78
VIII.	Future Work .....	79
	References.....	81

TABLE OF FIGURES

Figure 1 - VRAD/OpenCASCADE application [3]..... 10

Figure 2 - Trackball Parameters [5] ..... 11

Figure 3 - a) WIMP Interface of SolidWorks b) Virtual Testing Environment [6] ..... 12

Figure 4 - Hand Posture Database [7]..... 13

Figure 5 - Game Elements and Characteristics Applicable to CAD Improvement [9]..... 14

Figure 6 - Key Elements of a CAD Model Based Assembly Simulation System [11]..... 15

Figure 7 - YY3D Welcome Screen and Virtual Interior Scene Selector [12]..... 16

Figure 8 - CAD Model Conversion [13]..... 17

Figure 9 - Teamcenter Visualization Mockup Interface [14]..... 18

Figure 10 - Specifying Analysis Results and Type of Effects for Presentation [15] ..... 19

Figure 11 - ASDS Interface [16]..... 20

Figure 12 - Automatic Generation Process [18] ..... 21

Figure 13 - VR Setup for 3D Model Fault Identification Test..... 22

Figure 14 – Virtual Design II Grid Feature [20] ..... 23

Figure 15 - ENVIRON Interface [21] ..... 24

Figure 16 – COSTAR Cable Harness Design System [25]..... 26

Figure 17 - Two-Level Mapping of Virtual Assembly [30] ..... 27

Figure 18 - Soft Cable Assembly Routing in Virtual Environment [31] ..... 29

Figure 19 - Experimental Setup with PHANToM™ Omni [32].....	30
Figure 20 - Physical Workpiece and Virtual Workpiece in Coordinate Measuring Machine [33].....	31
Figure 21 – Web-based CAD User Interface [12] .....	33
Figure 22 - Holo-CAVE VR CAD Interface [6].....	35
Figure 23 - Physical (Voxel) Model, BREP (CAD) Model, and Tesselated (Game Engine) Model [17]..	39
Figure 24 - Soft Cable Simulation by (a) Segments, (b) Springs, and (c) Cosserat Model [31].....	40
Figure 25 - Unreal Engine: New Project - Virtual Reality.....	42
Figure 26 - Unreal Engine: VR Template.....	43
Figure 27 - Unreal Engine: Motion Controller Map.....	44
Figure 28 - Unreal Engine: VR Mode.....	45
Figure 29 - Unreal Engine: Settings.....	46
Figure 30 - Unreal Engine: World Outliner.....	47
Figure 31 - Unreal Engine: Export Selected.....	48
Figure 32 - Unreal: Home Menu.....	49
Figure 33 - Unreal: Snapping Menu .....	50
Figure 34 - Unreal: Insert Geometry Menu .....	51
Figure 35 - Unreal: Gizmo Menu (Translate, Rotate, and Scale) .....	52
Figure 36 - Google Blocks: Shapes Tool.....	53
Figure 37 - Google Blocks: Shape Creation .....	54

Figure 38 - MakeVR Pro: Files Menu .....	55
Figure 39 - MakeVR Pro: Tools Menu .....	56
Figure 40 - Blender Home .....	58
Figure 41 - FreeCAD "Part" Screen.....	59
Figure 42 - SolidWorks: Sketch Tab.....	60
Figure 43 - Unreal Airplane.....	63
Figure 44 - Unreal Quadcopter .....	64
Figure 45 - Unreal Predator .....	64
Figure 46 - MakeVR Pro Help Menu.....	66
Figure 47 - MakeVR Pro Airplane.....	67
Figure 48 - MakeVR Pro Quadcopter .....	67
Figure 49 - MakeVR Pro Predator .....	68
Figure 50 - Google Blocks Airplane .....	69
Figure 51 - Google Blocks Quadcopter .....	70
Figure 52 - Google Blocks Predator .....	70
Figure 53 - Blender Airplane .....	71
Figure 54 - Blender Quadcopter.....	72
Figure 55 - Blender Predator.....	72
Figure 56 - FreeCAD Start Page .....	73



Figure 57 - FreeCAD Airplane .....	74
Figure 58 - FreeCAD Quadcopter.....	75
Figure 59 - FreeCAD Predator.....	75
Figure 60 - SolidWorks Airplane.....	76
Figure 61 - SolidWorks Quadcopter .....	77
Figure 62 - SolidWorks Predator .....	78

LIST OF TABLES

Table 1 - Model Creation Times..... 62

## I. ABSTRACT

Computer Aided Drafting (CAD) is pervasive in engineering fields today. It has become indispensable for planning, creating, visualizing, troubleshooting, collaborating, and communicating designs before they exist in physical form. From the beginning, CAD was created to be used by means of a mouse, keyboard, and monitor. Along the way, other, more specialized interface devices were created specifically for CAD that allowed for easier and more intuitive navigation within a 3D space, but they were at best stopgap solutions. Virtual Reality (VR) allows users to navigate and interact with digital 3D objects and environments the same way they would in the real world. For this reason, VR is a natural CAD interface solution. Using VR as an interface for CAD software, creating will be more intuitive and visualizing will be second nature.

For this project, a prototype VR CAD program was created using Unreal Engine for use with the HTC Vive to compare against traditional WIMP (windows, icons, menus, pointer) interface CAD programs for the time it takes to learn each program, create similar models, and impressions of using each program, specifically the intuitiveness of the user interface and model manipulation. FreeCAD, SolidWorks, and Blender were the three traditional interface modeling programs chosen to compare against VR because of their wide-spread use for modeling in 3D printing, industry, and gaming, respectively. During the course of the project, two VR modeling programs were released, Google Blocks and MakeVR Pro; because they were of a similar type as the prototype software created in Unreal Engine, they were included for comparison as part of this project. The comparison showed that the VR CAD programs were faster to learn and create models and more intuitive to use than the traditional interface CAD programs.

## II. INTRODUCTION

In the past, VR gear was only attainable by the very rich, by big businesses, or by the government, but lately the cost and availability of VR headsets and capable computer hardware are becoming attainable for the average consumer. Both Oculus Rift and HTC Vive, while still rather expensive for consumer devices, are now readily available and easy to use. Also, many free game engines such as Unity, Cryengine, and Unreal Engine have abundant training materials available and are able to create VR content. As VR technology develops, it will inevitably become less expensive and easier to use, making it more attractive, practical, and obtainable to a greater number of people. Using VR as an interface device for creating digital media is not a new concept; there is much software of this kind freely available now. The dilemma with most of this software is that it is underdeveloped compared to its non-VR counterparts. Fully featured digital media creation software, including CAD, can take years for teams of a hundred or more to complete and bring to market. For that reason, this project will attempt to create a VR interface for existing CAD software or create proof-of-concept VR CAD software.

## III. LITERATURE REVIEW

“VRMDS: an intuitive virtual environment for supporting the conceptual design of mechanisms” by Juan Alvarez and Hai-Jun Su is a paper presenting a VR program the authors developed called Virtual Reality Mechanism Design Studio (VRMDS) [1]. The software allows users to create jointed line segments within the virtual environment to simulate wires, cables, chains, etc. and perform dynamic simulations on them. The authors describe design choices they made in the

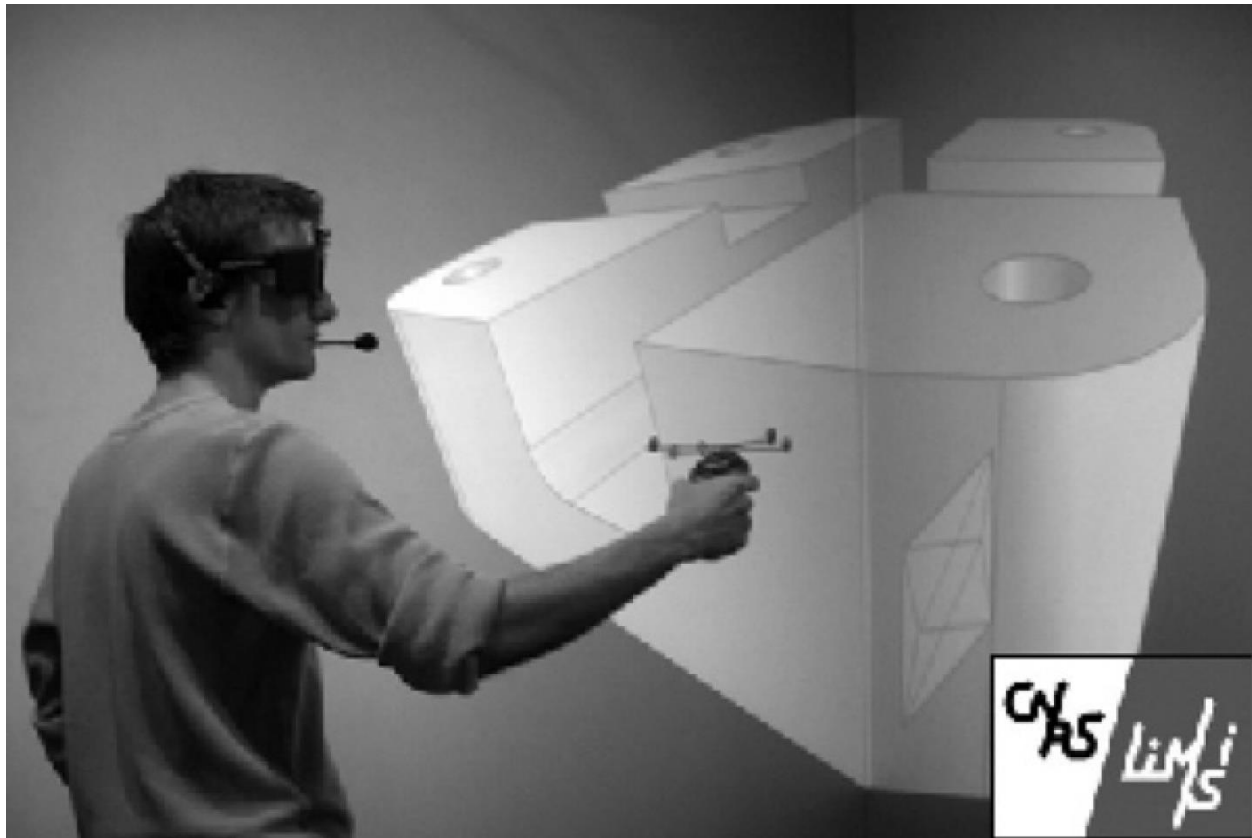
process of creating the software including the types of joints used to connect the line segments and the use of MATLAB's SimMechanics for simulation.

“Projects in VR” by Julien Berta is an editorial about using VR for visualization in Catia 5 [2]. Dassault Systems partnered with a company called Fakespace to develop support for VR and other immersive display devices. The method used for this is file conversion from CAD to either a VR specific file format or VRML (Virtual Reality Modeling Language). Some of the down-sides of the conversion process were loss of geometry precision, loss of topology data, loss of dimensions, units, names, constraints, relationships, and IDs, and loss of kinematic behaviors. They found that, because of the data loss, the conversion process could take as long as several weeks in order to manually rebuild the lost information. Three solutions were identified:

1. Set up efficient gateways to ease the conversion process.
2. Derive a common file format for both CAD and VR software.
3. Integrate VR technologies in to the core of the CAD software.

“VR-CAD integration: Multimodal immersive interaction and advanced haptic paradigms for implicit edition of CAD models” by P. Bourdot, et al. presents an approach to integrate VR and CAD in order to allow direct editing of 3D models within VR [3]. The method that they focused on allowed direct editing of b-rep models (short for boundary representation – these are the type of models used in typical CAD programs). They developed a program for VR capable of editing and creating CAD models that they called VRAD (Virtual Reality Aided Design) using OpenCASCADE. During testing, they found that interaction with CAD models was limited by

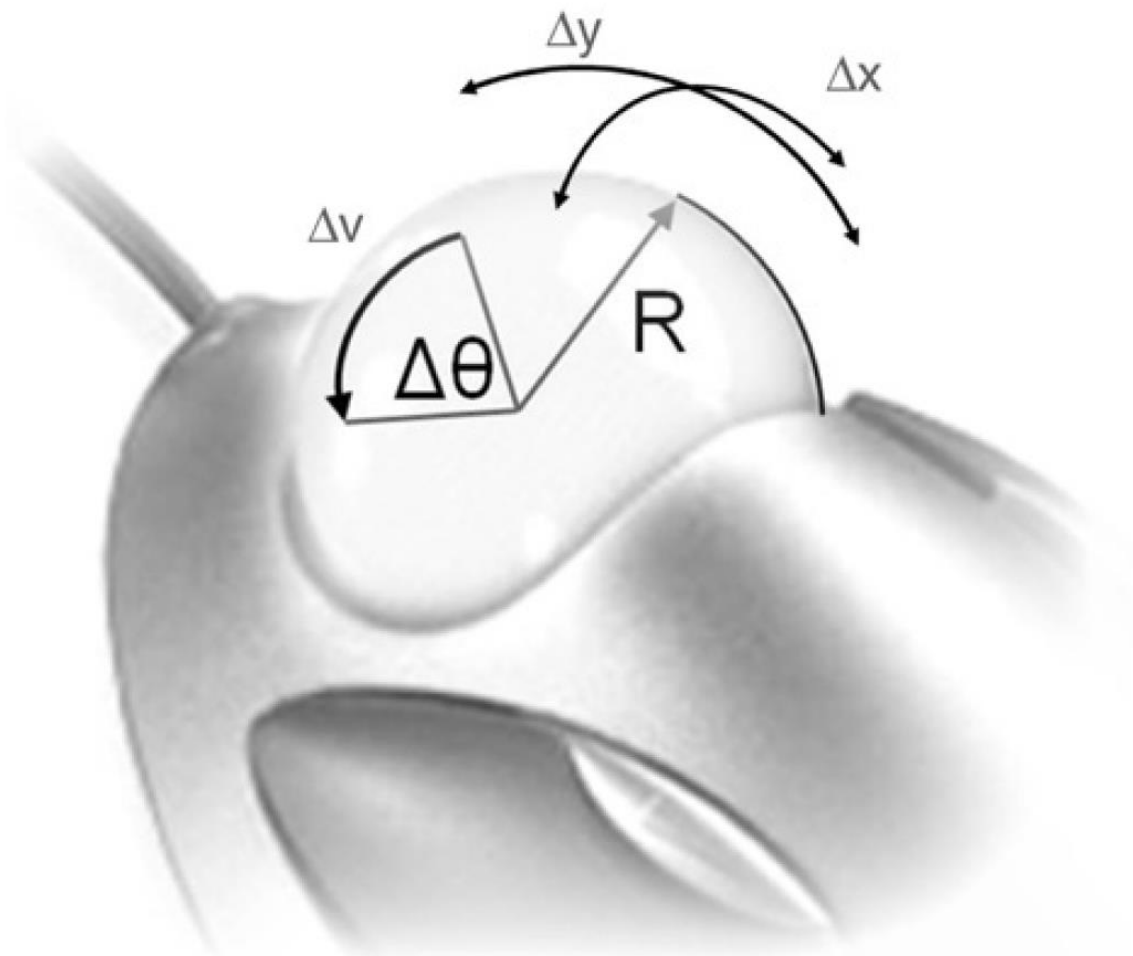
poor accuracy of hand gesture controls. Drawbacks of this system include cost and availability of the necessary hardware, development time, and implementation time including time it takes to learn the system and modify it to suit the user.



*Figure 1 - VRAD/OpenCASCADE application [3]*

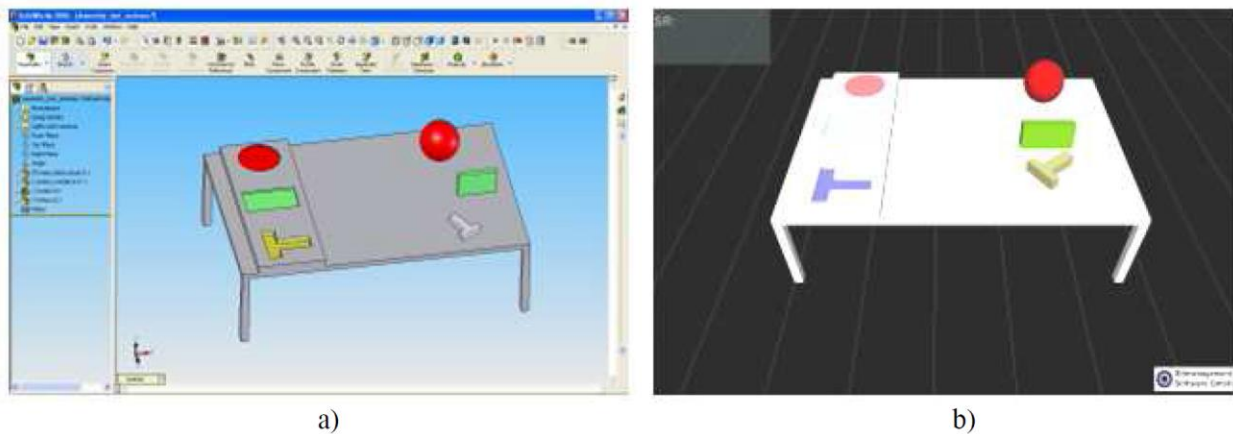
“Design considerations for the design of an advanced VR interface for knowledge management and its relevance to CAD” by Ahmad Eshaqa and Peter Karboulonis is about a knowledgebase visualization algorithm developed by them to be used with VR with potential applications for CAD [4]. Since CAD library trees can become extremely complex, it is useful to have a way to search the tree by topic. This paper explains the algorithms and other methods used to search databases from within a VR environment.

“Improving bi-manual 3D input in CAD modelling by part rotation optimisation” by M. Fiorentino, et al. is a research paper about the benefits of using a trackball for part rotation in CAD [5]. 92% of their test subjects indicated that using the trackball for part rotation was “easier, more comfortable and better overall” than other methods. Several methods were used to map part rotation to trackball movement; non-linear rotation was preferred to the other methods because fast trackball movements were usually made to indicate that the user desired the model to move quickly at the expense of precision.



*Figure 2 - Trackball Parameters [5]*

“Evaluation of CAD model manipulation in desktop and multimodal immersive interface” by Florin Girbacia is a paper about evaluating the use of multimodal Virtual Reality for CAD model manipulation [6]. For the evaluation, the author used a Holo-CAVE system (a large-scale multi-wall projection system and optical tracking system to track the user’s position and orientation) with data gloves for gesture recognition and voice recognition for input commands to determine the viability of replacing the tradition WIMP (windows, icons, menus, pointer) interface. The author found that task completion time in VR was dramatically faster than WIMP times because of the intuitive and direct nature of model manipulation in VR.

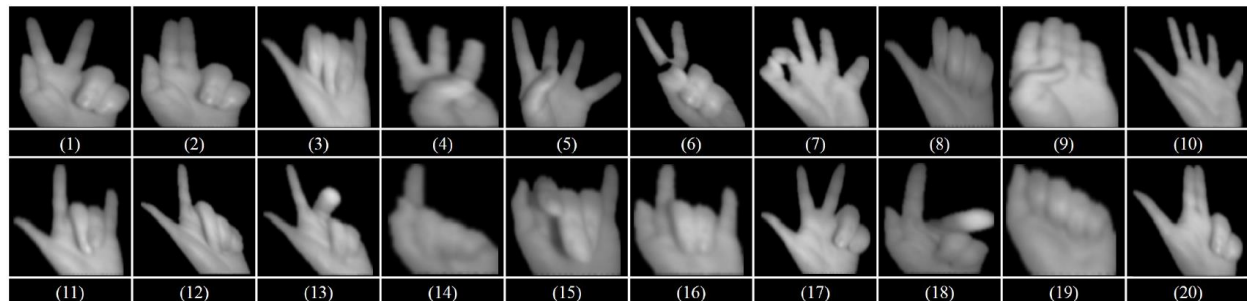


*Figure 3 - a) WIMP Interface of SolidWorks b) Virtual Testing Environment [6]*

“Intuitive Interface for Conceptual 3D Modeling” by Jinmiao Huang is a dissertation on the creation of a framework CAD program for creating conceptual 3D models from gestural interaction [7]. The author used a 2D display and the Leap Motion controller to capture a user’s hand posture for command input and hand position and orientation for model translation, rotation, and scaling. Research showed that testing subjects preferred the gesture-based modeling interface to a traditional WIMP interface and the author acknowledged that the use of 2D display was



unintuitive stating that a virtual reality system would be useful to assure the right placement of models.



*Figure 4 - Hand Posture Database [7]*

“Application of VR Technology linked with 4D CAD System for Construction Project” by Leen-Seok Kang, et al. is a research paper about using CAD models to visualize construction sites at key time schedule milestones in a VR environment [8]. They used VRML to visualize 4D models created using AutoCAD and 3D Max. During the course of their research, they found that using VR for visualization improved the effectiveness of 3D and 4D simulation.

“Engineering design using game-enhanced CAD: The potential to augment the user experience with game elements” by Zoe Kosmadoudi, et al. is an article investigates how the future development of CAD systems in engineering could benefit from gaming techniques and mechanisms with a focus on graphical user interfaces (GUIs), specifically how they could make CAD more engaging and intuitive [9]. The author found that, in particular, CAD could benefit from 3D visualization in VR and mobile tablets for intuitive and natural interaction with models and for real-time capturing of the model creation process for use in training or to help identify

undesirable actions. Also, game GUIs with contextual menus could make the CAD model creation process more enjoyable and intuitive experiences.

Games elements and characteristics		Limited CAD aspects
Game mechanics	Process of use; gameplay Goals defined by game world & system's performance feedback Variable difficulty level (e.g. for score keeping); GUI has successive layers of complexity	Results of the processes Goals defined by task requirements & Manager's performance reports No difficulty levels; GUI organised on design tasks
Aesthetics and the Graphical User Interface (GUI)	Different representation tools and customise UI elements graphical richness Randomness & Humour Use of communications strategies: Introduce new information when knowledge is incomplete and inconsistent Metaphors	Limited customizability (if available) & no multiple integrated representations - Limited assistance and constructive feedback
Games technology	Variety of interactive devices Continuous innovation in content and control; game engines	- Force feedback devices Development in designing product visualisation (e.g. VR)
User experience	Variety of experiences emotionally appealing cognitive models; metaphors Mood & Motivation	Experiences not consistent Functionality but not usability

*Figure 5 - Game Elements and Characteristics Applicable to CAD Improvement [9]*

“Comparative Study of CAD Software, Web3D Technologies and Existing Solutions to Support Distance-Learning Students of Engineering Profile” by Zona Kostic, et al. is a research paper about using Web3D to teach distant learners how to use various CAD programs and to collaborate on CAD model creation projects [10]. The authors chose five commonly used CAD programs for their research: SolidWorks, Inventor, CATIA, Pro/ENGINEER, and AutoCAD. They illustrated the collaborative process they used for each program to convert native file formats to Web3D format and weighed the pros and cons of each. SolidWorks was found to be the easiest to master of all the processes followed by CATIA for its ease of converting to and from the requisite file formats.

“CAD model based virtual assembly simulation, planning and training” by Ming Leu, et al. is a research paper on assembly simulation using CAD systems in virtual environments [11].

Additionally, the paper describes methods to gather surface data of a 3D object and how to use the data to create CAD models, as well as methods for motion capture, force and sound modeling, and multi-modal (graphic, haptic, and sound) rendering. The authors present compelling scenarios for using VR and AR for assembly planning and training applications and using CAD model systems for a more efficient product design cycle, product quality improvement, and skill enhancement through virtual training.

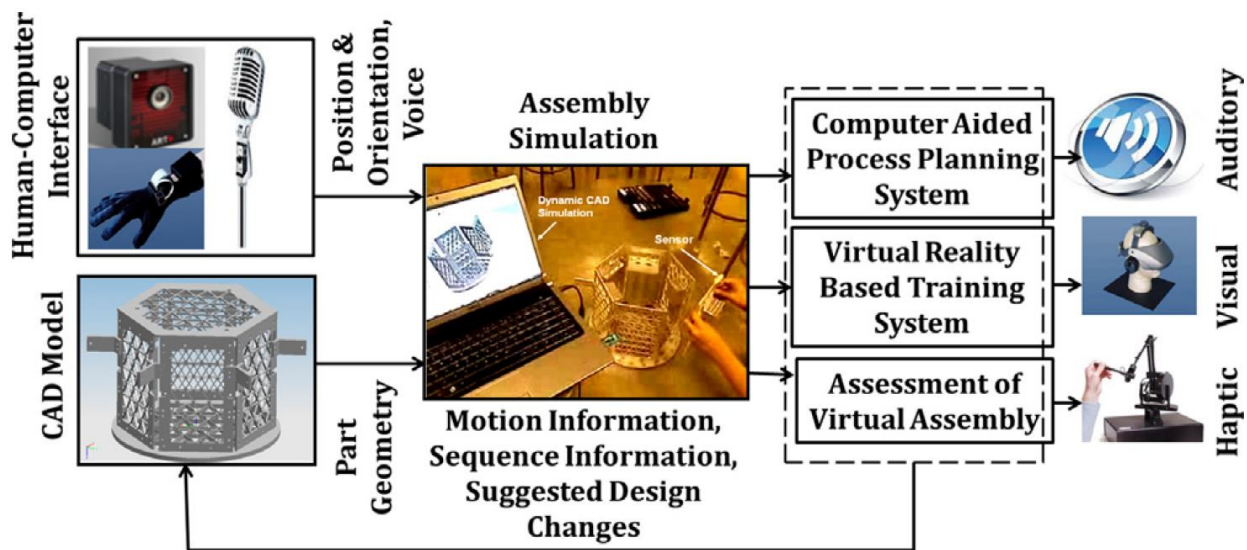


Figure 6 - Key Elements of a CAD Model Based Assembly Simulation System [11]

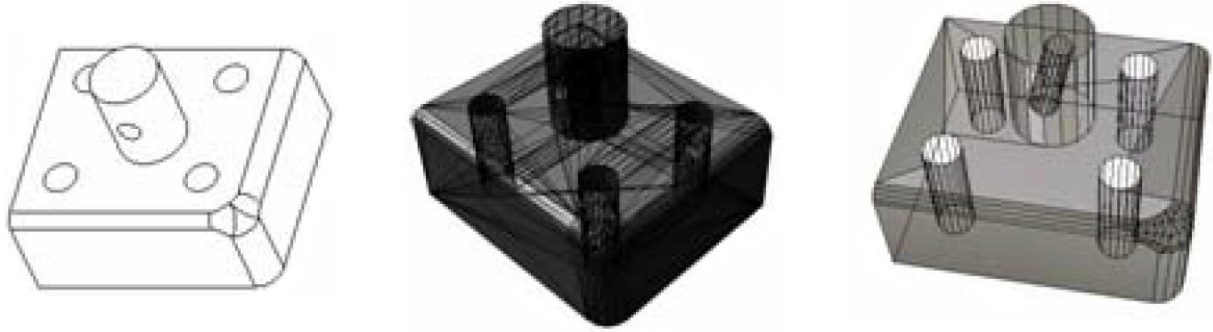
“Using Virtual Reality to Improve Design Communication” by Chao Liu is a Master’s thesis about creating a proof-of-concept web-based virtual reality program called YY3D for clients to communicate interior designs [12]. The author converted CAD models created in AutoCAD, 3ds Max, and SketchUp into the X3D file format to share as 3D graphics in the YY3D virtual environment over the internet. Users would then orient models within YY3D to express their design requirements. Although the software developed for this thesis focused primarily on interior

design, the author recognized the potential for this method to benefit many other collaborative creation projects including building design and other architectural design projects.



Figure 7 - YY3D Welcome Screen and Virtual Interior Scene Selector [12]

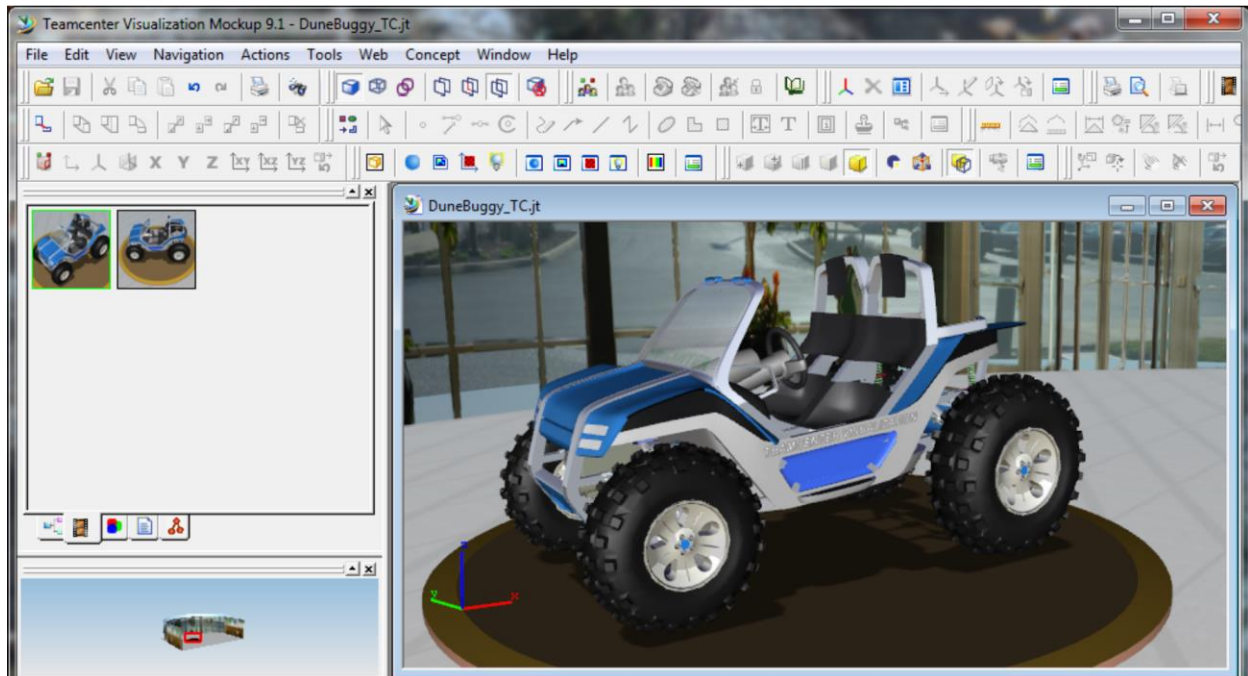
“Key Technology Research on Data Conversion Between B-Rep Based CAD and VR Model” by Xianguang Luo, et al. is an article about how to efficiently convert from b-rep CAD models to polygonal models using an algorithm that they developed [13]. The CAD software that they used was Pro/E and they used Pro/Toolkit to extract surface data from the models they created. The surface data they extracted was fed into their algorithm in order to create points in 3D space for polygonal models. They then compared the resulting models produced by their algorithm with models produced by another conversion program, PolyTrans; their models had the same level of detail but more efficient way of controlling the number of triangles used.



(a) CAD b-rep model (b) PolyTrans conversion (c) algorithm conversion

*Figure 8 - CAD Model Conversion [13]*

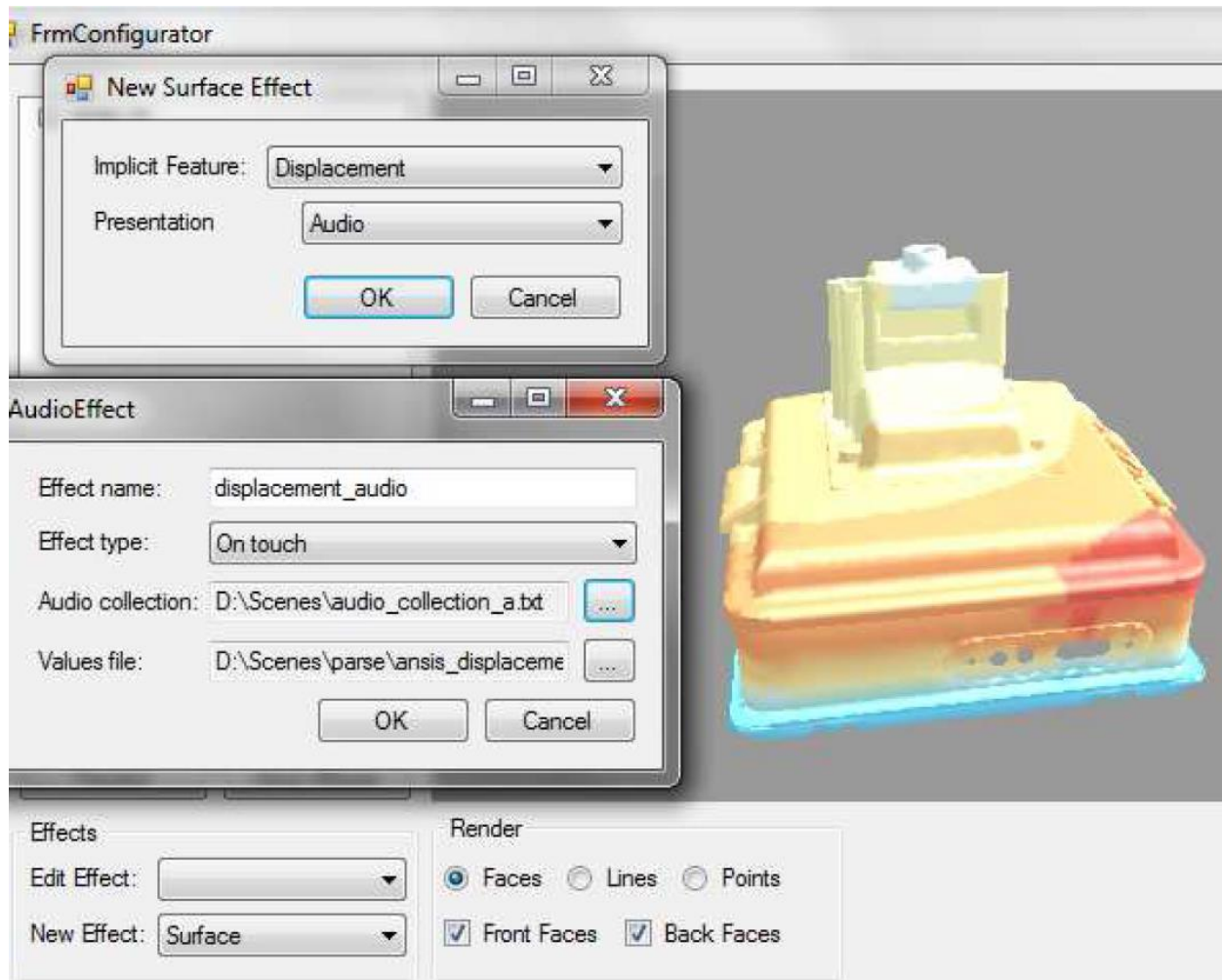
“Natural user interfaces for interdisciplinary design review using the Microsoft Kinect” by Anastacia Maria MacAllister is a Master’s thesis about creating a method of interacting with CAD models developed using Teamcenter Visualization Mockup<sup>TM</sup> with head tracking and hand gestures captured by a Microsoft Kinect<sup>TM</sup> to facilitate rapid prototyping and collaboration [14]. The author found that model creation using hand gestures were more intuitive than traditional mouse and keyboard interaction and saved time for complex tasks, but gestures were not the most efficient solution in every situation, specifically simple tasks where features are not buried in menus.



*Figure 9 - Teamcenter Visualization Mockup Interface [14]*

“Post-processing of Engineering Analysis Results for Visualization in VR System” by Stoyan Maleshkov and Dimo Chotrov is a research paper about a software program developed to convert and display CAD models and finite element analysis (FEA) data in a virtual environment [15]. The application developed by the authors converts CAD models into a VRML file format and incorporates FEA results in several ways including surface mesh color mapping, sound representation, and haptic representation. They found that using their method significantly improved user perception and understanding of FEA results especially when evaluating multiple analysis results simultaneously as a result of the software’s multimodal presentation capability.





*Figure 10 - Specifying Analysis Results and Type of Effects for Presentation [15]*

“A system for rapid creation and assessment of conceptual large vehicle designs using immersive virtual reality” by Christian Noon, et al. is a research paper where the authors assess a program they developed called the Advanced System Design Suite (ASDS) to create and test the feasibility of conceptual design prototypes using metamodeling (FEA, CFD, etc.) approximations [16]. Pro/CONCEPT was used to create prototype models and then converted using a system called ENVIRON (Environment for VIRTUAL Objects Navigation) to a simplified model for analysis in

ASDS. Using ASDS, many models could be created and evaluated quickly. Because of this, designers had additional time to create models that pushed the boundaries of design decisions.

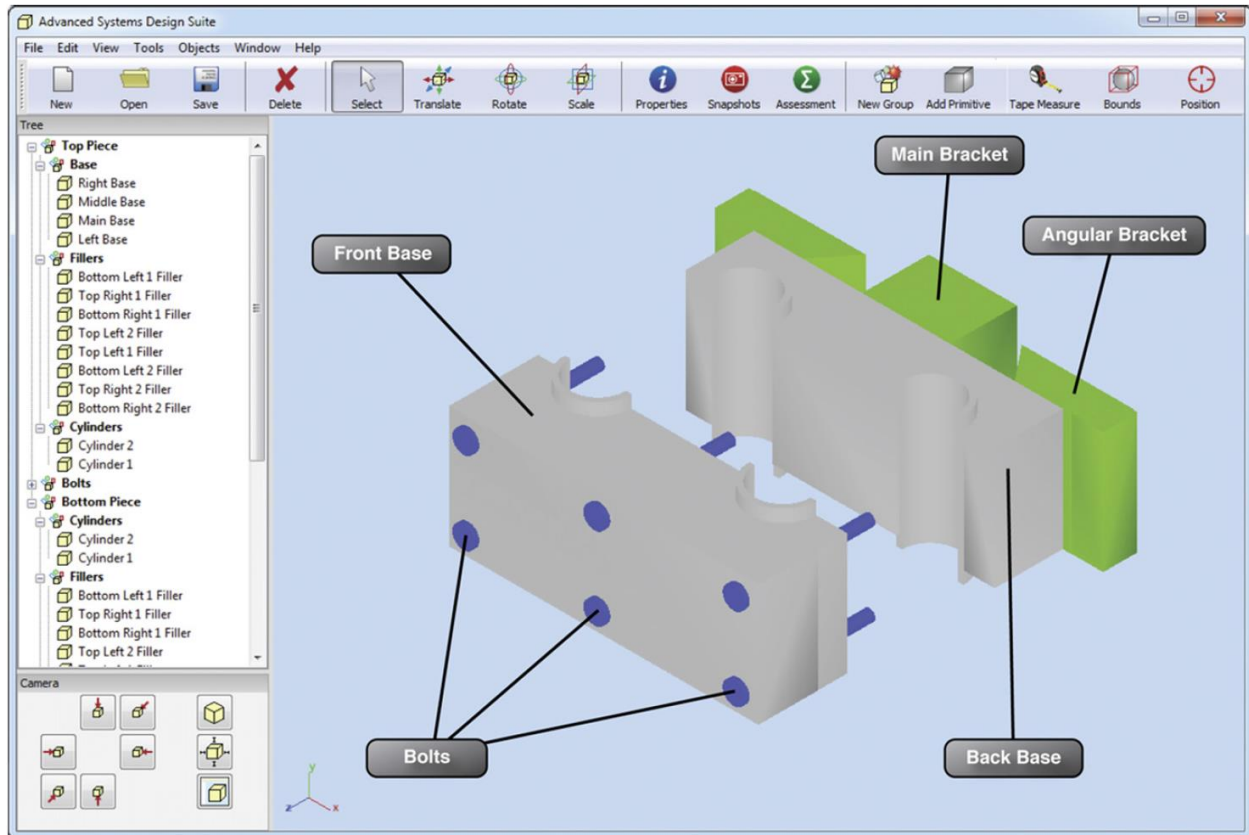


Figure 11 - ASDS Interface [16]

“Enabling natural Interaction for virtual reality” by Ryan Pavlik is a dissertation about software and methods to support an accurate reflection of interactions with real objects within a virtual environment [17]. For the project, the author developed software to use a Wii remote as a head-tracking device and a software platform called VR JuggLua for rapid development of natural interactions in VR. Due to the extensive research and development that went in to this project, the author was able to recommend numerous insightful ways that VR could be improved.



“A Methodology for Reducing the Time Necessary to Generate Virtual Electric Substations” by Alexandre Carvalho, et al. is a paper about an automated process established to streamline the process of converting 2D floor plans and 3D CAD models into 3D virtual environments [18]. The authors developed an algorithm to identify the type and orientation of equipment at electrical substations from an image file of the floor plan and then construct a 3D environment by converting CAD models to polygonal models and arranging them in the correct position. They found that their method reduced generation time by 83%.

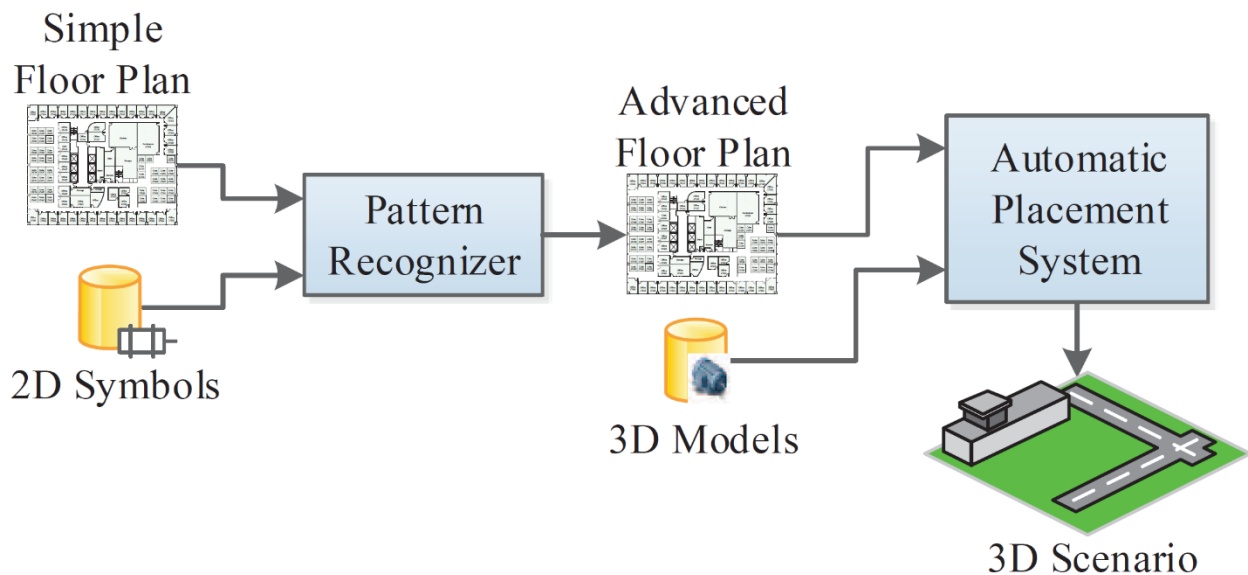


Figure 12 - Automatic Generation Process [18]

“Immersive Virtual Reality Prototype for Evaluating 4D CAD Model” by Lucky Pratama is a Master’s thesis about using 3D geometry to visualize in virtual reality how a construction project will progress over time [19]. The author developed a VR visualization program using Cryengine 3 and used Oculus DK1 and DK2 VR headsets for visualization. Several tests were performed to determine the effectiveness of identifying faults in 3D models in VR versus a traditional interface; participants were able to identify faults in VR almost twice as quickly versus the traditional

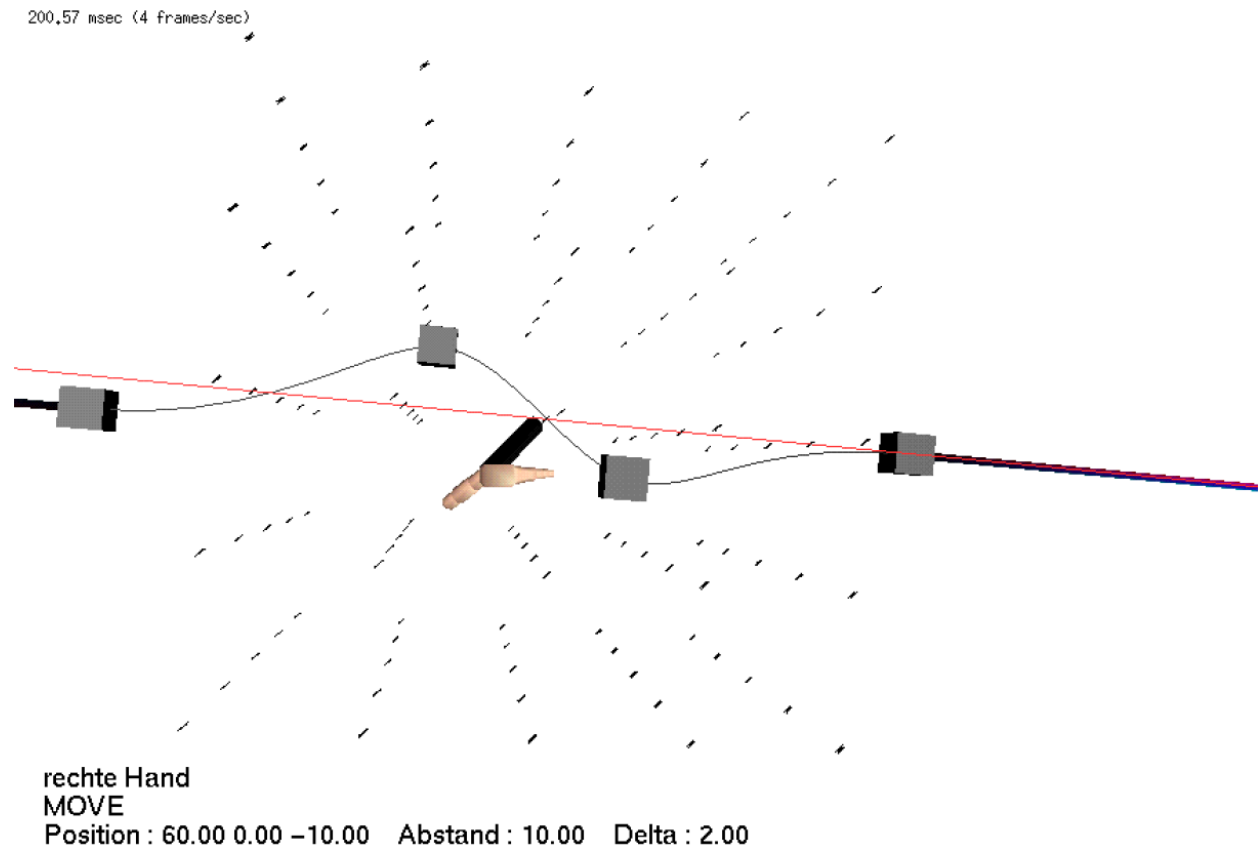
interface, however, the VR computer hardware and headsets were below what was necessary for comfortable interaction and participants reported discomfort due to the VR experience.



*Figure 13 - VR Setup for 3D Model Fault Identification Test*

“Integration of Basic CAD Functions into a VR Environment” by Ralf Rabätje is a paper about the authors impressions of a VR CAD system called “Virtual Design II” [20]. In the software, the user has the ability to create and edit CAD files directly, however, the author identified two disadvantages: the user did not know his location or orientation with the virtual environment or the real one and the input devices lacked accuracy. To address the problem of orientation, the author suggested implementing a grid feature within the virtual environment. Consequently, HTC Vive address both of these issues: headset and controller tracking has sub-millimeter accuracy

and there are several positional awareness solutions incorporated including an outward-facing camera in the headset, a grid feature, and a choice of several visual “close proximity to the user-defined boundary-wall” alerts.



*Figure 14 – Virtual Design II Grid Feature [20]*

“Environ: Integrating VR and CAD in Engineering Projects” by Alberto Raposo, et al. is an article about using an application they developed called ENVIRON (Environment for VIRTUAL Objects Navigation) to convert and visualize CAD models in VR [21]. A plugin developed for ENVIRON converts complex CAD models to a format viewable in VR by using several rendering optimization techniques including far voxels, specialized CAD GPU primitives, and reverse

engineering of GPU primitives. The application allows users to manipulate models from within the virtual environment, however, it does not allow the modifications to be saved in CAD file format.



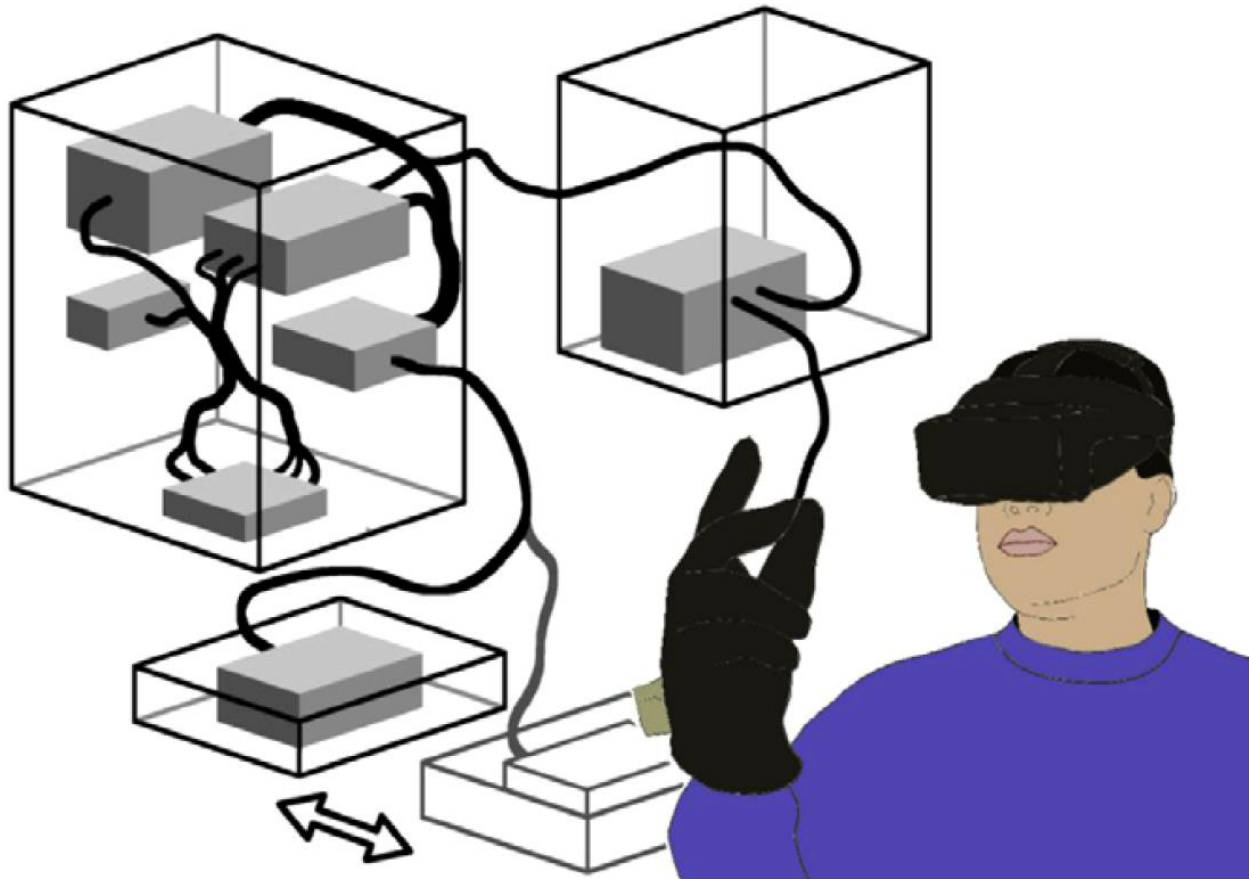
*Figure 15 - ENVIRON Interface [21]*

“A solution to Integrate CAD/CAE Systems and VR for Complex Product Design Processes” by SiQun MA and Zhi-Li GAO is a project paper about using VRML for collaborative design of 3D models [22]. The authors used several software programs in their workflow: initial models were created using CATIA, CosmoWorlds was used to setup light and camera positions, and then models were saved in VRML file format to be shared and viewed online. Using this process enables collaborative rapid prototyping from any location with an internet connection.

“Towards hybrid modelling environments – Merging desktop-CAD and virtual reality-technologies” by R. Stark, et. al is a research paper on the feasibility and human factors of CAD within VR [23]. The authors researched promising features of virtual environments, potential

advantages of immersive modeling, necessary criteria for immersive modeling including a user analysis, and potential stumbling blocks. They also performed a case study using a VR-cave (a VR environment using rear projected walls and shutter glasses for stereoscopic vision) to investigate benefits of a VR design process; they found that eleven out of fourteen participants preferred the VR-cave to paper-and-pencil. The main disadvantages that they found were as a result of the state of the technology at the time – tracking was imprecise and computer hardware did not produce high enough frame rates.

“Automated design process modelling and analysis using immersive virtual reality” by Raymond Sung, et al. is a research paper about several ways to record the workflow of experts in order to catalog the process for use in training [25]. Amongst other methods, the authors use VR to capture and replay design and assembly processes of cable harnesses using a system called COSTAR (Cable Organisation System Through Alternative Reality). Their research showed that using VR for creation has the added benefit of easily capturing the process for future use by trainees or to easily determine in the future what part of the process produced an undesirable result.



*Figure 16 – COSTAR Cable Harness Design System [25]*

“Evaluation of a commodity VR interaction device for gestural object manipulation in a three dimensional work environment” by Frederick Thompson is a Master’s thesis about using a LEAP Motion Controller for gestural control to manipulate CAD and CAE models in the 3D work environment of the Advanced Systems Design Suite (ASDS) developed by researchers at Iowa State University [26]. Testing showed that gestural control was not significantly difficult to learn and improved execution time over mouse interaction. Performance of tool selection gestures revealed that gestures were not an ideal replacement for toolbar buttons; the author found that the best use of gestural control was with model interactions in three or more axes.

“Virtual Reality Company, ImmersaCAD, Unveils New Website, App” is an article in the Journal of Engineering about an app for Google Cardboard and Samsung GearVR called ImmersaCAD designed to visualize and navigate CAD models in virtual reality [29]. The software uses an intuitive navigation system that claims to solve the problem of motion sickness typically associated with virtual reality. Users simply press one button to move forward in the direction they are looking from within the virtual environment.

“A review of virtual reality and haptics for product assembly (part 1): rigid parts” by PingJun Xia, Antonio Lopes, and Maria Restivo is a research article on using VR and haptics for product assembly from rigid parts [30]. In the article, the authors provide a comprehensive overview of VR and haptics used for product assembly, including hardware and software. They review results of testing including the useful and undesirable characteristics of each VR product assembly system and method as well as provide recommendations where the technology could be improved.

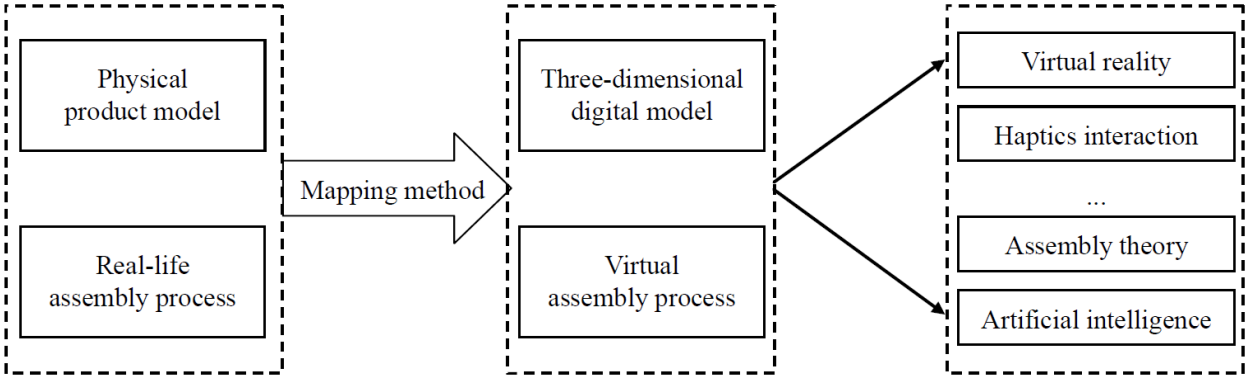
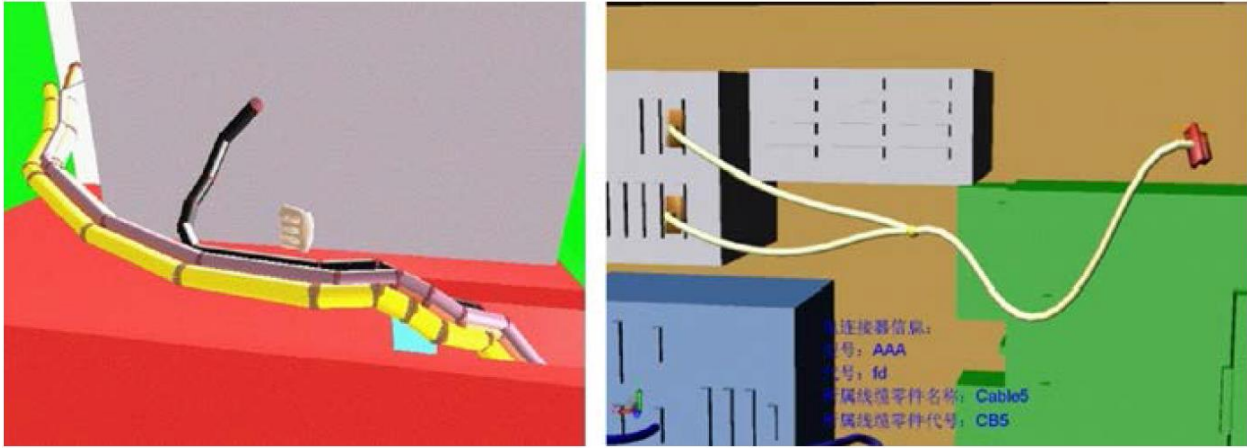


Figure 17 - Two-Level Mapping of Virtual Assembly [30]

“A review of virtual reality and haptics for product assembly: from rigid parts to soft cables” is part 2 of the previous research article; it provides a comprehensive overview of VR and haptics software and hardware used for soft cable routing and assembly including methods of physics simulation of the cables [31]. The authors identified several barriers to widespread application of virtual assembly technology at the time of their research in three categories: device factors, technology factors, and validation factors. Device factors included high cost and heavy weight of head mounted displays (HMDs) and haptic devices, low resolution and brightness of HMDs, and data glove reliability and accuracy issues. The technology factors included no universal CAD file format, computational cost of collision detection, and inaccurate physics modelling and haptics interaction. Validation factors also prevent practical application of virtual assembly; many VR assembly systems are not based on an assembler’s real needs – they instead align with expectations of design engineers and are used primarily for geometry visualization and to capture assembly procedures for training purposes. Finally, the authors identified five aspects of potential future research directions:

1. CAD interface standardization and model light-weight representation.
2. Haptics interaction for rigid parts and soft cables.
3. Physics modeling for large-scale complex products.
4. Knowledge and AI for product assembly.
5. Psychology and cognition modelling for virtual assembly design and training.





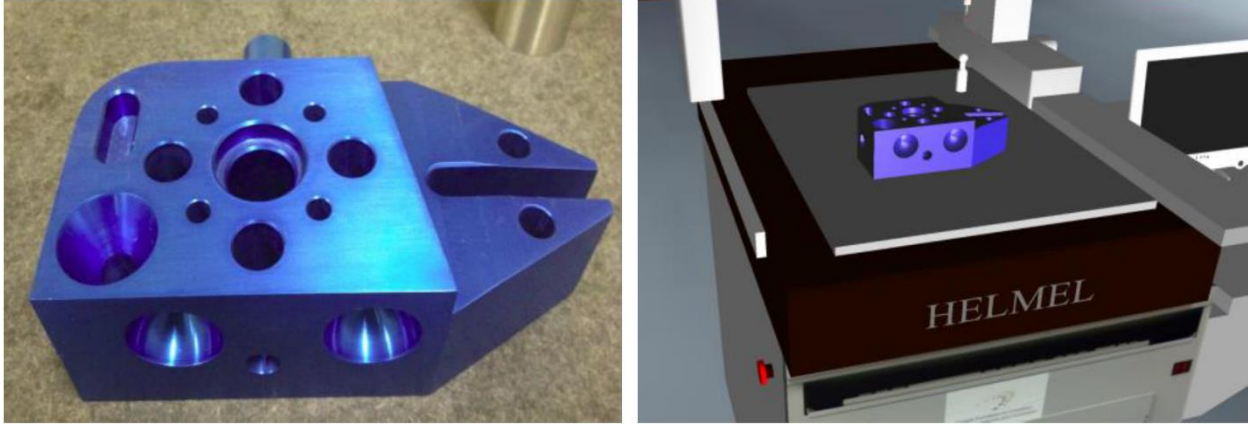
*Figure 18 - Soft Cable Assembly Routing in Virtual Environment [31]*

“The effect of sub-threshold forces on human performance in multi-modal computer-aided design” by Mehrdad Zedeh, et al. is a research paper about increasing accuracy of hand movements in virtual environments by using path predicting and haptic device force feedback [32]. The authors’ experimental setup involved using a PHANTOM<sup>TM</sup> Omni as an input device to determine what types of forces on participants’ hands caused them to achieve the desired result with the highest rate of success. Experimentation indicated that forces below perceptible levels towards a target goal helped participants reach their intended goal and opposing forces helped them stay on the final target.



*Figure 19 - Experimental Setup with PHANToM™ Omni [32]*

“Development of a VR-based CMM System for Industry Training and CMM Path Planning” by Long Zhao is a Master’s thesis on the creation and uses of a virtual Coordinate Measuring Machine (CMM) [33]. The most obvious use identified by the author is for training purposes for new users to become familiar with CMM operations and experienced users to verify their inspection path and avoid damaging the real CMM due to incorrect operation. The author proposed several methods of collision detection in a virtual environment that could improve the accuracy simulating collision physics of virtual objects.



*Figure 20 - Physical Workpiece and Virtual Workpiece in Coordinate Measuring Machine [33]*

#### IV. PRELIMINARY RESEARCH

Studying material related to VR and CAD has been very encouraging regarding the goals of this project. It appears that no other research has accomplished what this project intends to achieve, and many sources indicate that the goals established for this project are the logical next steps in their own research [1] [6] [7] [9] [14] [17] [19] [30]. The ultimate goal regarding CAD within VR is to allow intuitive and direct 3D editing on native CAD models within VR [1] [6] [7] [14] [30].

VR provides an immersive and intuitive environment to work in; users can interact with huge 3D models of complex products to appreciate their scale, structure, and spatial orientations to optimize design, reduce or eliminate mockup fabrication, shorten development, reduce cost and risk, guide and train assembly workers, and improve assembly quality and efficiency [30]. Within VR, users interact with models and the environment as they would in reality, so interactions are visceral and intuitive, which require less training, occur more quickly and naturally, and are a better representation of what the final product will be like in the real world. That is, after all, the ultimate

purpose of designing models in CAD – to create them in the physical world, or at the very least to simulate the physics and aesthetics of what the model would be like in physical form.

Many sources found that collaborative CAD development is also very useful, stating that collaborative virtual environments (CVEs) are perhaps one of the most promising applications of VR [11] [12] [14] [16]. Several methods were employed for collaboration including the use of X3D to view and modify CAD models within a web browser (figure 21) [10] [12], taking advantage of networking capabilities within game engines to create CVEs that multiple VR users could occupy simultaneously [9] [19], and using universal save file formats to allow development across various CAD platforms [10] [11] [19]. The ability to collaborate with other developers during conceptual design proved to be particularly useful for Christian Noon, et al. [16]; early design stages typically use many engineers to produce a concept design pool with many different simple models from which the best ideas proceed to detailed design. Having a simple tool that allows for fast creation and collaboration is invaluable to this process.

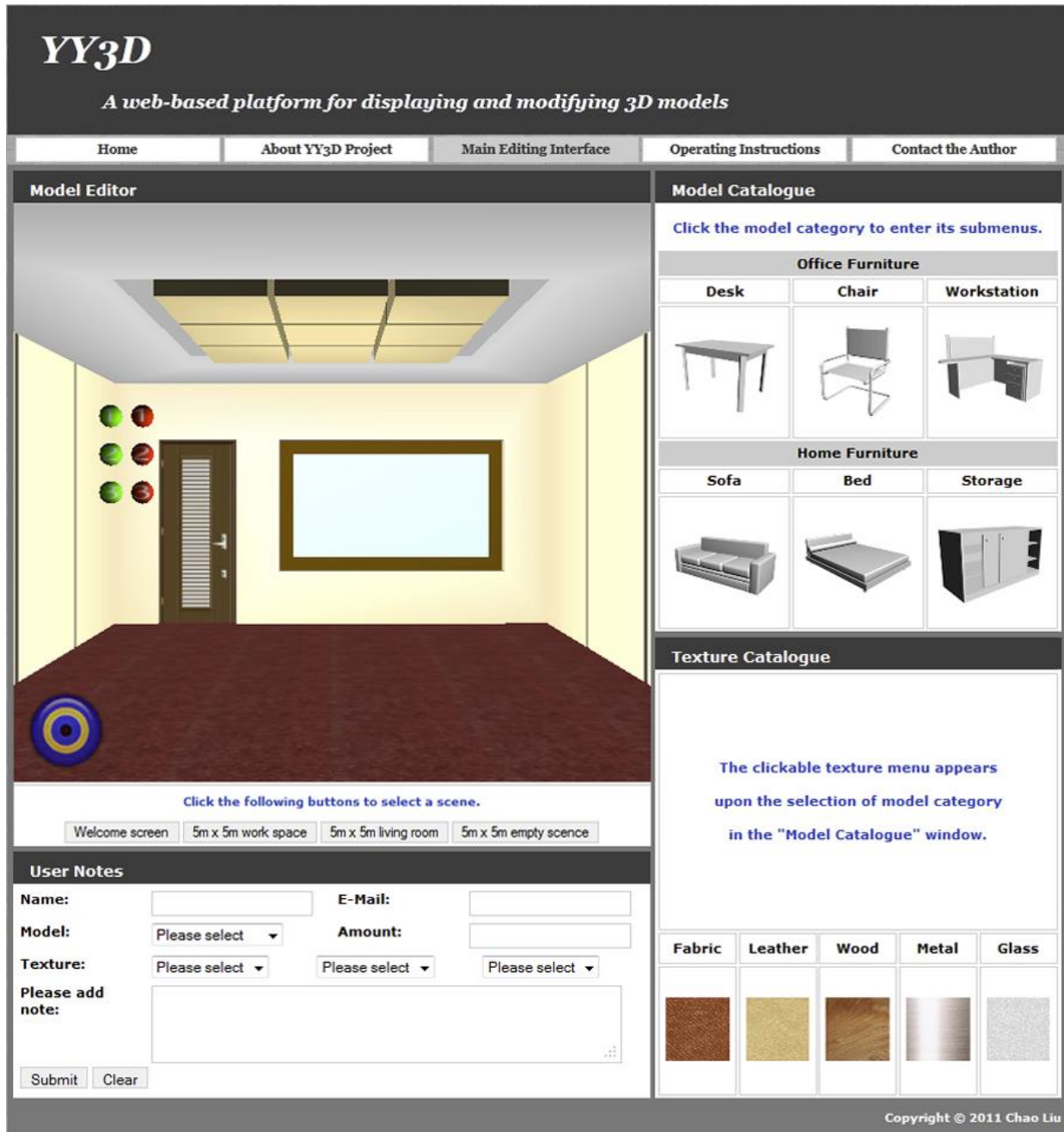


Figure 21 – Web-based CAD User Interface [12]

Model creation was typically faster and more intuitive with a VR interface. Florin Gîrbacia used a large-scale multi-wall projection system called Holo-CAVE, data gloves, and voice recognition

to interface with the CAD program SolidWorks; it was observed that the direct immersive interface dramatically improved the speed and ability to manipulate CAD models because of better perception of depth and natural translations and rotations [6]. Frederick Thompson found that gestural interaction has the greatest advantage for rotation, translation, and scaling of CAD models in 3D (16.9 seconds for mouse vs 2.1 seconds for LEAP motion controller to rotate) [26]. Jinmiao Huang used a 2D display and Microsoft Kinect (a motion-sensing camera originally designed for the Xbox 360 and redesigned for Xbox One game consoles) for gesture recognition to interface with three CAD platforms: Maya, 3ds Max, and SketchUp. He found that three modeling experts were able to create models in half the time or less [7]. One source found that using gestures to create CAD models was not always faster – Anastacia MacAllister discovered in her research with Microsoft Kinect that using gestures to trigger CAD commands, while more intuitive, may not reduce completion time for simple CAD tasks [14].



*Figure 22 - Holo-CAVE VR CAD Interface [6]*

Developing software is a time-consuming process, especially from the ground up or with inexperienced programmers. Two sources recognize that game engines allow developers to focus on content rather than details of implementation and that they can be used for purposes other than games [9] [19]. Using game engines to develop a virtual environment comes with the advantages of having a built-in physics engine, network support including the ability to develop for multiplayer, pre-existing object models, and user interfaces that make development easier to start. Cryengine, Unity, and Unreal Engine were identified as the most capable game engines for free development of a VR CAD interface or VR CAD prototype software based on the research done

by Kosmadoudi, et al., Lucky Pratama, and developer documentation indicated compatibility with the HTC VIVE VR system for each of the aforementioned game engines [9] [19] [24] [27] [29].

A reliable and bug-free user experience is crucial for software used for work; losing work due to software instability or save file corruption is inconvenient and, in many cases, unacceptable. Save file format and software stability is key when developing CAD software [10] [11] [19]. Both Zona Kostic, et al. and Luck Pratama stress the importance of CAD software stability stating that if the environment is flawed it can be a source of significant risk [10] [19]. Errors introduced into the CAD model's save file could cause corruption which would lead to incorrect parameters in the model or introduce errors that would cause the model to load incorrectly or not at all. Zona Kostic, et al. demonstrates that the X3D file format is best for displaying CAD models within a web browser and also is a robust lightweight format for saving basic CAD model data [10]. Long Zhao explains how Eon Studio is a virtual environment (VE) development tool that provides primitive models (cubes, spheres, cylinders, etc.) that are used to represent imported models from other CAD programs; the models are managed using a tree structure hierarchy to define spatial relationships and can be manipulated (but not created) within virtual environments [33]. Lucky Pratama explains that Cryengine's visual programming tool was used to create his prototype VR CAD interface and that Cryengine was particularly unstable. He and Ming Leu, et al. mention that Cryengine required elaborate effort to translate CAD models into a format that could be viewed within the engine, often requiring reorientation and scaling [11] [19]. Ming Leu, et al., whose research is from 2013 (2 years before the release of HTC VIVE and Oculus Rift consumer VR), states that in order to improve realism in virtual assembly, the following technologies should be developed:

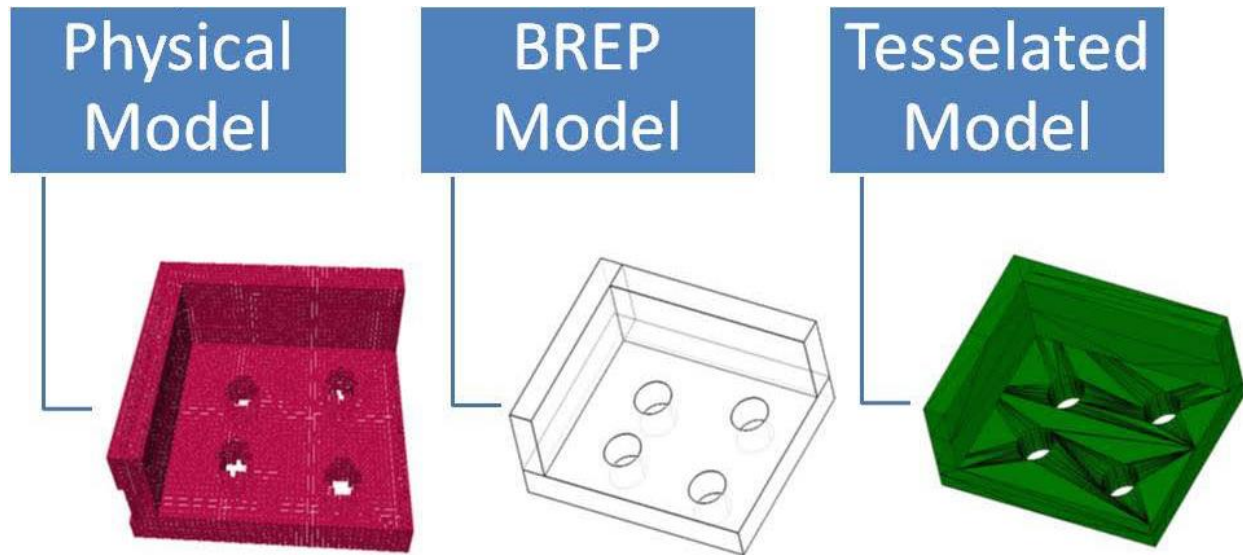


1. High-fidelity dynamic graphic displays, specifically increased resolution and robustness and reduced weight, latency, and cost.
2. More accurate, efficient, robust, and low-cost motion tracking.
3. Haptic devices that can generate both tactile and force stimuli.
4. Less invasive and more intelligent sensors that can capture information beyond motion, such as human emotions.
5. Algorithms and software to represent, organize, and manipulate large sets of CAD model data to reduce latency and increase refresh rate in graphic, haptic, and auditory displays for more realistic VR [11].

With the HTC VIVE, most of these goals are achieved. VIVE has high resolution, high field of view, and robustness as well as reduced weight, low latency, and low cost compared to previous generations of VR head mounted displays (HMDs). VIVE has sub-millimeter accuracy lighthouse tracking sensors that track both the headset and hand-held motion controllers. The VIVE's motion controllers can generate haptic vibration but do not generate force stimuli. Latency and refresh rate of the graphic, haptic, and auditory displays are in the appropriate ranges to be imperceptible by the user, which is one of the most important factors to reduce fatigue and virtual reality motion sickness. Virtual reality motion sickness can also arise when unexpected or unnatural motion of the user occurs in VR; developers have discovered several ways to alleviate this problem including using teleportation, grasping stationary objects and pulling oneself through the VR environment, or moving only in the direction that the user is facing [29].

In order to display models in VR, they must have polygonal geometry to ensure fast frame rates; CAD models are more precise and must be translated in order to view in VR [11] [14] [15] [16] [17] [19]. In order to translate CAD models to VR, texture maps should be used to represent the detail of their surfaces, primitive solids such as boxes, spheres, and cylinders should be used to represent objects in the distance, increasing their level of detail as the user moves closer, billboards should be used to represent complex objects in the far distance, and objects should be selectively loaded depending on the viewpoint [19]. There are also several tools available that were discussed in the source material that convert CAD data to VR: the ENVIRON (Environment for VIRtual Object Navigation) system decreases file size, enhances material properties, and tessellates complex surfaces to produce a VR representation of CAD models, MEMPHIS (Middleware for Exchanging Machinery and Product Data in Highly Immersive Systems) is a data exchange system for integrating product data management (PDM), CAD, and VR through common interfaces, and Pro/CONCEPT is a stripped down software product for modeling product ideas without unnecessary complications or constraints [16]. In traditional CAD software, models are displayed using parametric boundary representations (b-reps), which are mathematical representations of the faces and edges that make up an object model. Because of the way physics is simulated in VR, displaying CAD models with b-reps would result in low performance. The way physics simulation is typically performed is by using small cube-shaped volume elements called voxels. Voxels are used to simulate collisions by detecting when the voxels from one model come in close proximity to the voxels of another model. Because voxels are cube-shaped, rounded surfaces are only approximated by their use, so features that should mate according to b-rep data, such as round holes and pegs will not fit together unless specific conditions are added to the software to allow

for such a scenario. To simulate object mating, it is common to use voxels and b-rep data in hybrid methods in addition to or instead of "snap to" [17].



*Figure 23 - Physical (Voxel) Model, BREP (CAD) Model, and Tessellated (Game Engine) Model [17]*

Physics modelling of deformable objects (such as cables) in VR is a challenge because of the balance between accuracy and performance efficiency [31]. There are several methods used to model cables in physics engines, all of which use a one-dimensional centerline and related cross-section; the cables are considered unstretchable, unshearable, with the cross-section consistent and undeformable. The most basic is by the use of segments with mass at the intersections, whose location are represented with three spacial coordinates and orientation of each segment represented by a quaternion (mathematical notation to represent orientation and rotation in 3D) [31].

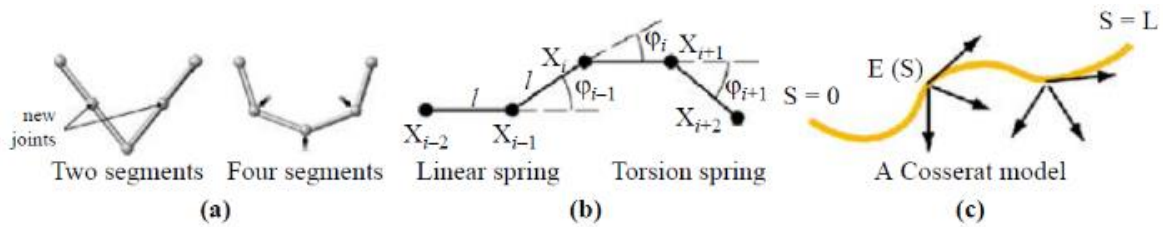


Figure 24 - Soft Cable Simulation by (a) Segments, (b) Springs, and (c) Cosserat Model [31]

Based on the information discovered during this literature review, creating a direct VR interface for existing CAD software is not possible because the performance required to simulate the physics of b-rep models in real-time is too great for even the best computer hardware available at the current time. However, creating a VR CAD program using a game engine is a viable option. Unreal Engine is the only engine that has VR development mode, native VIVE support, and dynamic tessellation [24] [27] [28]. Cryengine and Unity are both compatible with VIVE, but do not have native support or documentation on how to develop applications for the VIVE. Because of this, Unreal Engine's reported reliability and ease of use, and the reported unreliability of Cryengine and Unity, Unreal Engine will be used to create a prototype VR CAD program for this project.

The information gleaned from previous research was invaluable. Ryan Pavlik states that future research should apply the substantial base of existing research and software to new advances in conventional VR hardware [17]. Because of recent advancements in consumer VR, applying lessons learned from the available references will do exactly that; it is an obvious next step to build on the work done by previous VR/CAD related research projects.

## V. METHOD

The original plan for this project was to create a single VR CAD program within Unreal Engine and compare it with several traditional interface CAD programs; several things happened during the course of the project that made it seem best to modify the plan. As I was learning to use Unreal Engine, two VR modeling programs were released that, while not as robust as some traditionally interfaced CAD programs, allow the creation of 3D models. Because they are both capable of producing similar models to the program that I created using Unreal Engine, I decided to include them for testing in this project. These two programs are Google Blocks and MakeVR Pro.

While learning how to create content within Unreal and becoming familiar with the user interface, creation methods and workflow, I was pleased to discover that Unreal included VR game creation templates which greatly facilitated the speed of development for my prototype CAD program. From within VR in Unreal, users can access all of Unreal's game creation tools and use them to create 3D models. For the purpose of this project, I created a custom level within Unreal, became familiar with the menus needed to create models using primitives, and disregarded the menus that were not necessary for model creation.

The prototype VR CAD program was created using the following procedure. Upon starting Unreal Engine, the “Projects” and “New Project” tabs are visible (figure 25). To begin, a new project was started using a blueprint template within Unreal by selecting the “New Project” tab, “Virtual Reality”, and clicking “Create Project”.

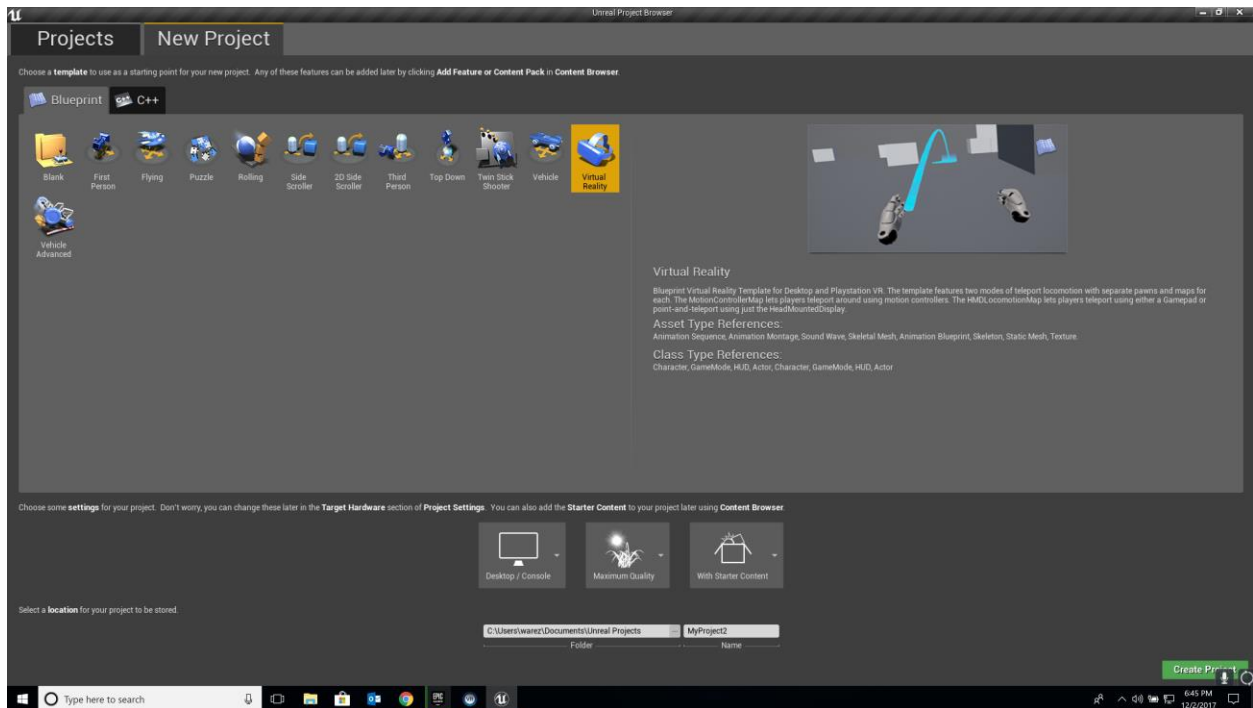


Figure 25 - Unreal Engine: New Project - Virtual Reality

The Unreal Engine VR template includes several starter maps and assets to facilitate new project creation. When a new VR project is created, Unreal starts with a map called “VR Template” meant to prompt the user to first select the type of controller (gamepad or motion controllers) and key bindings to use by default (figure 26).

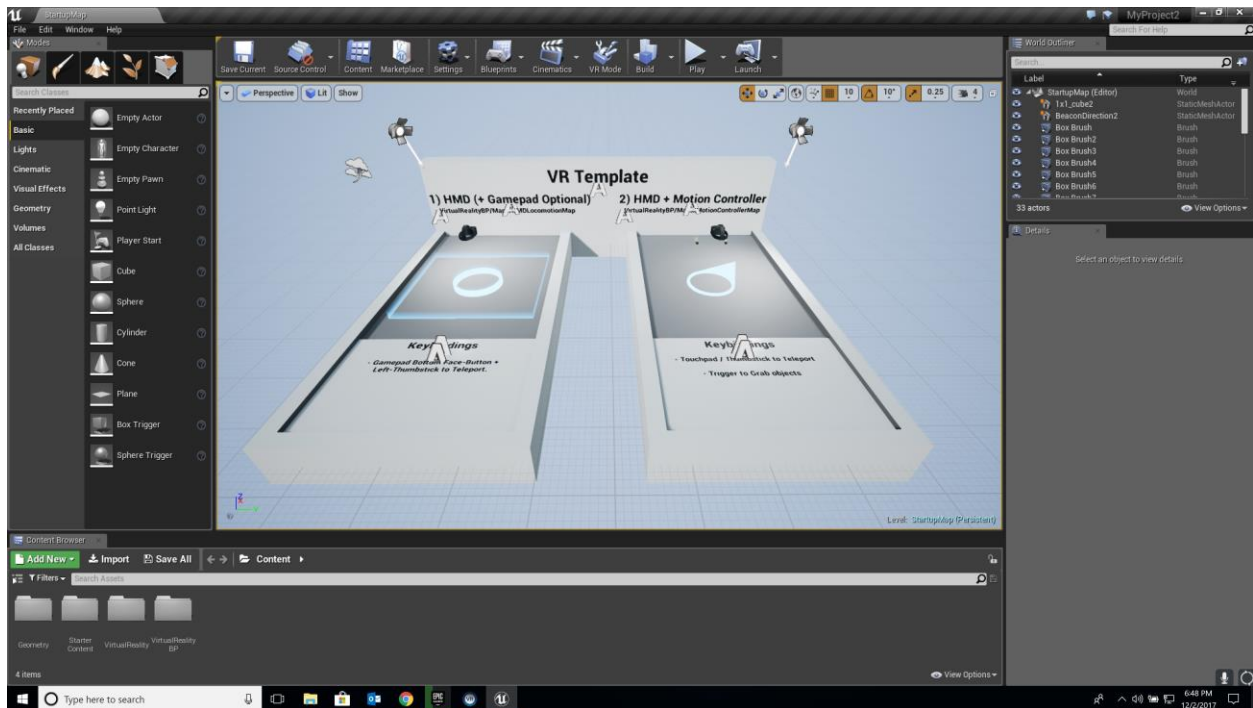


Figure 26 - Unreal Engine: VR Template

To open a template map compatible for use with motion controllers, click “File” and “Open Level” and then double click “Motion Controller Map” from within VirtualRealityBP, Maps (figure 27).

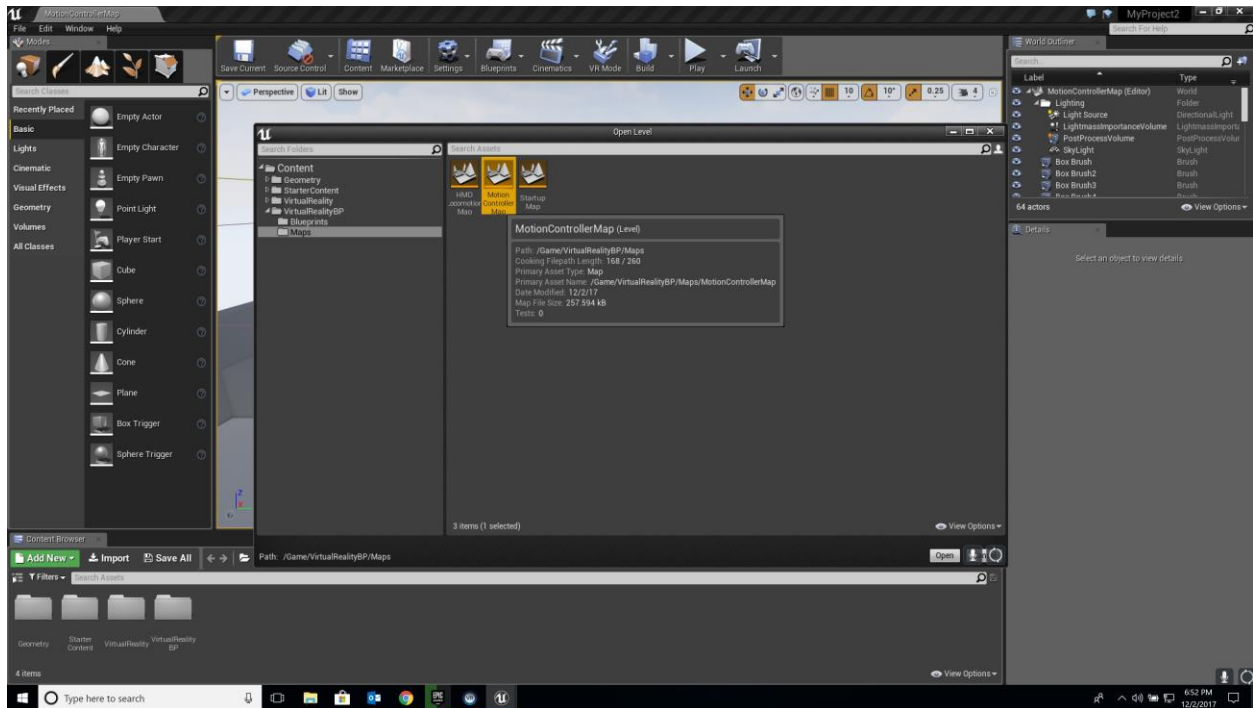


Figure 27 - Unreal Engine: Motion Controller Map



Upon opening the Motion Controller Map, the template level contains starter, “example” content to demonstrate physics interactions including lighting, collision detection, and gravity. At this point, the user can enter VR creation mode to modify the template from within VR by clicking “VR Mode” or remain in the traditional Unreal interface to make changes to the level.

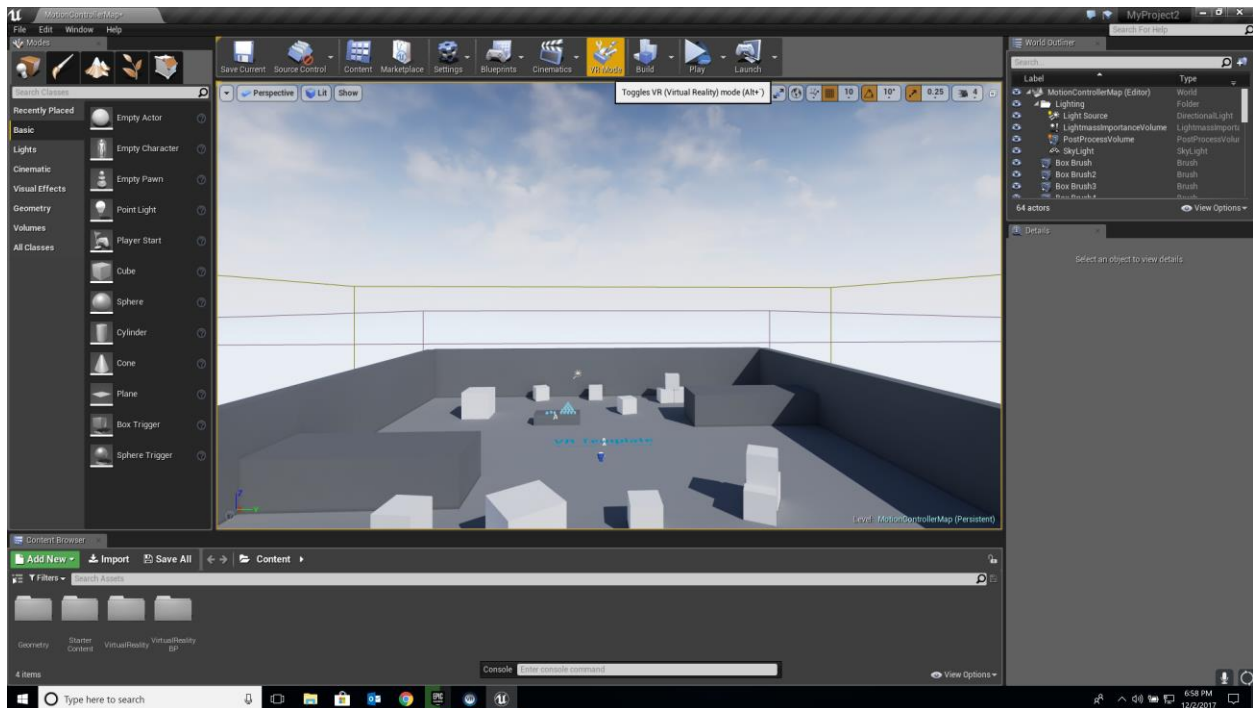


Figure 28 - Unreal Engine: VR Mode

From the “Edit” menu, selecting “Editor Preferences”, and “VR Mode” allows the user to modify and simplify the VR user interface according to taste; during model creation for this project, default settings were used (figure 29).

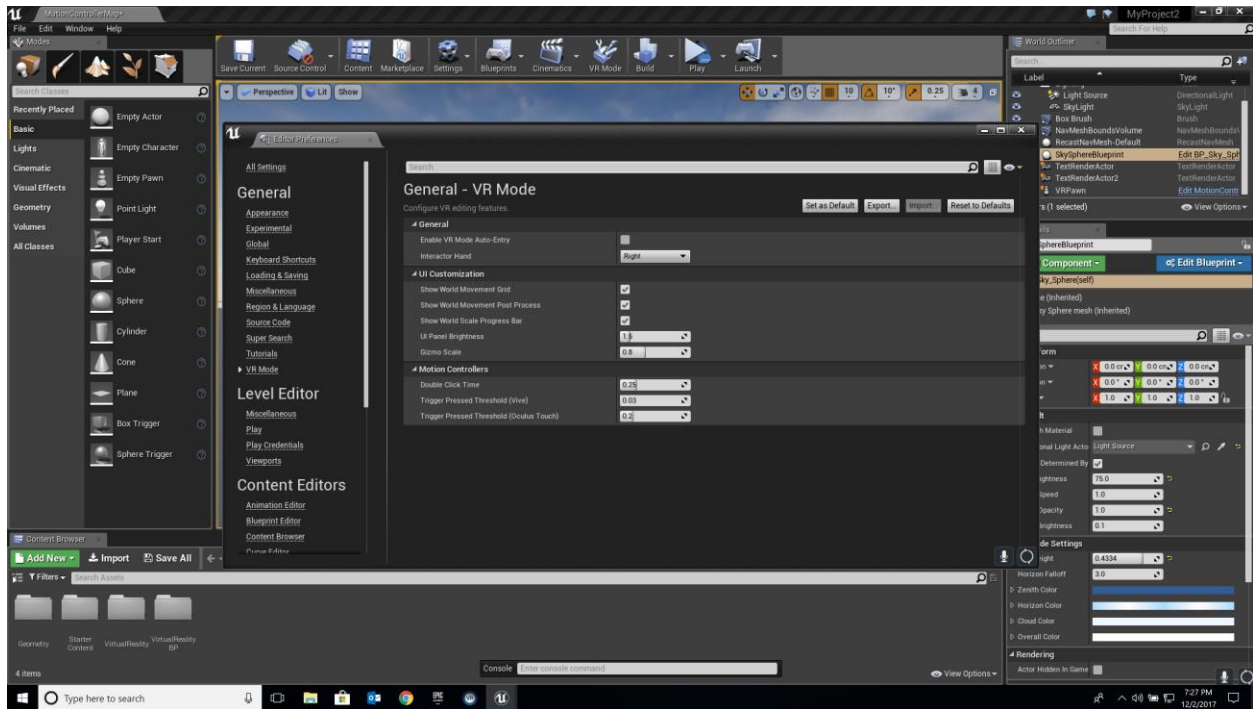


Figure 29 - Unreal Engine: Settings

In order to create an empty virtual environment for model creation, remove example content from the template level. Select unnecessary content from the main viewport window and press Delete, ensuring that associated content is also deleted from the “World Outliner” window (figure 30); select associated content from the list and press Delete.

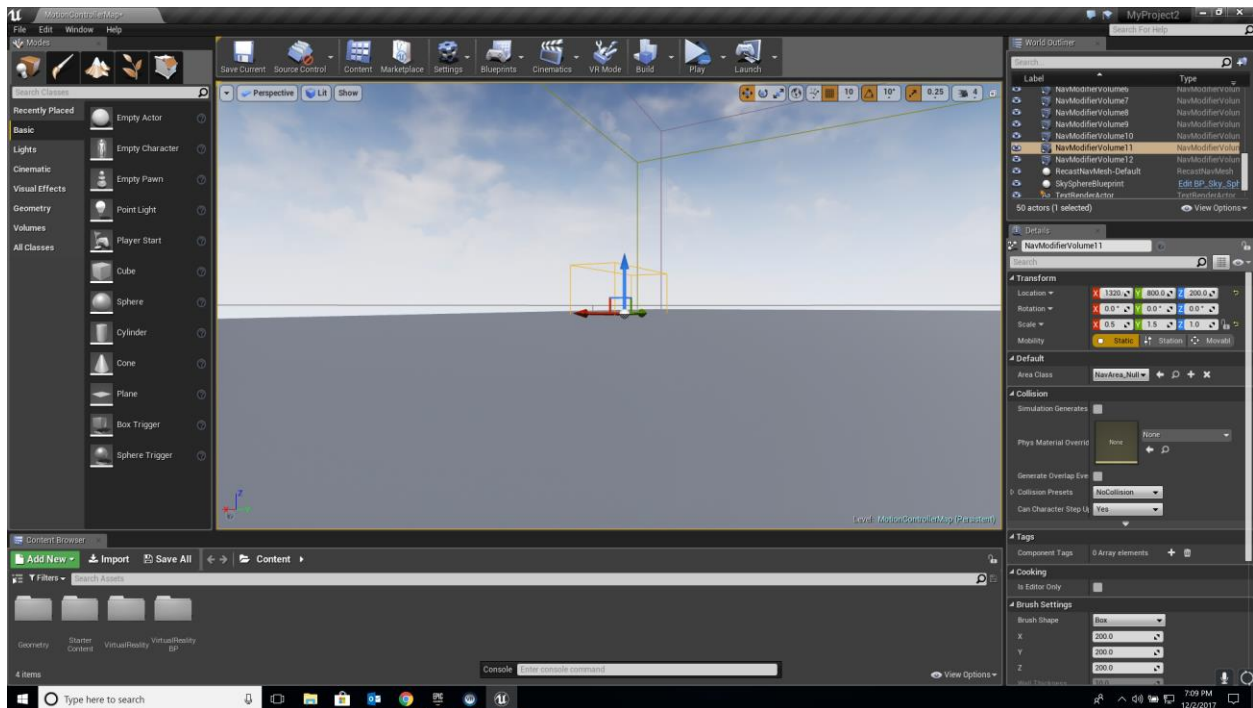


Figure 30 - Unreal Engine: World Outliner

To save the empty level as a starting point for model creation from within VR, click “File” and “Save Current”. At this point, models created from within this level can be selected and saved as several common file formats by selecting the model and clicking “Export Selected” from the “File” menu (figure 31).

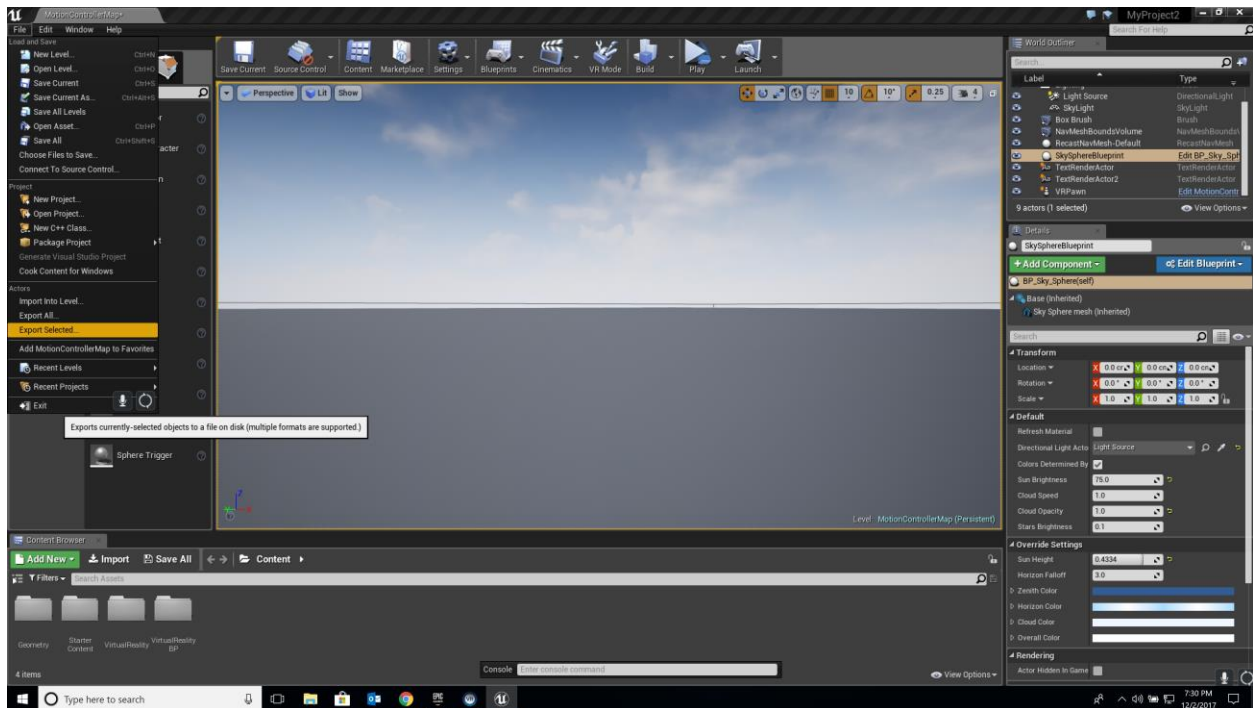
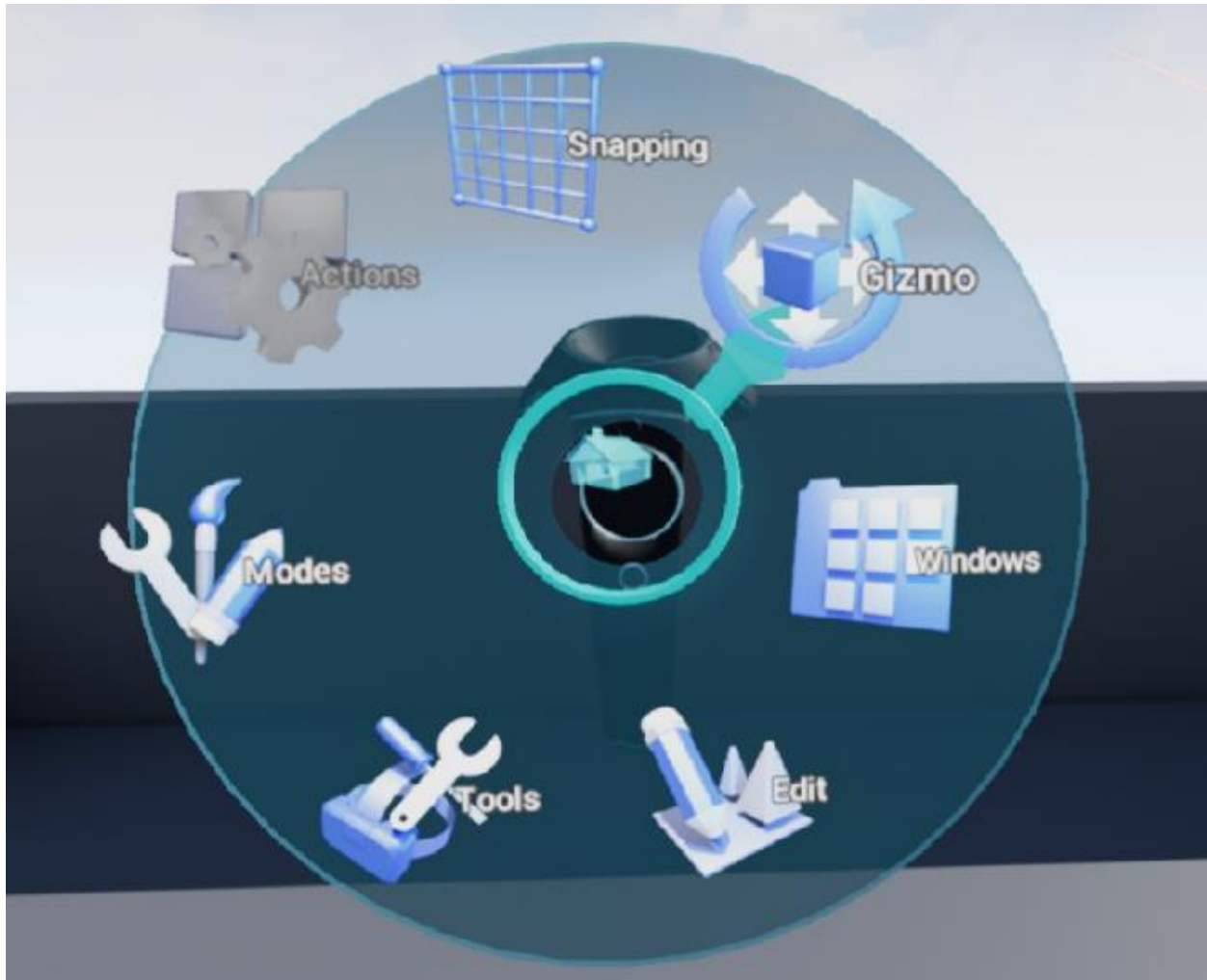


Figure 31 - Unreal Engine: Export Selected

Workflow for model creation in the prototype Unreal CAD program was accomplished using the following steps:

1. Activate the Unreal VR home menu from the left-hand motion controller (figure 32)



*Figure 32 - Unreal: Home Menu*

2. From the “Snapping” menu, set snapping options for translation, rotation, and scaling (on/off and increment – see figure 33)

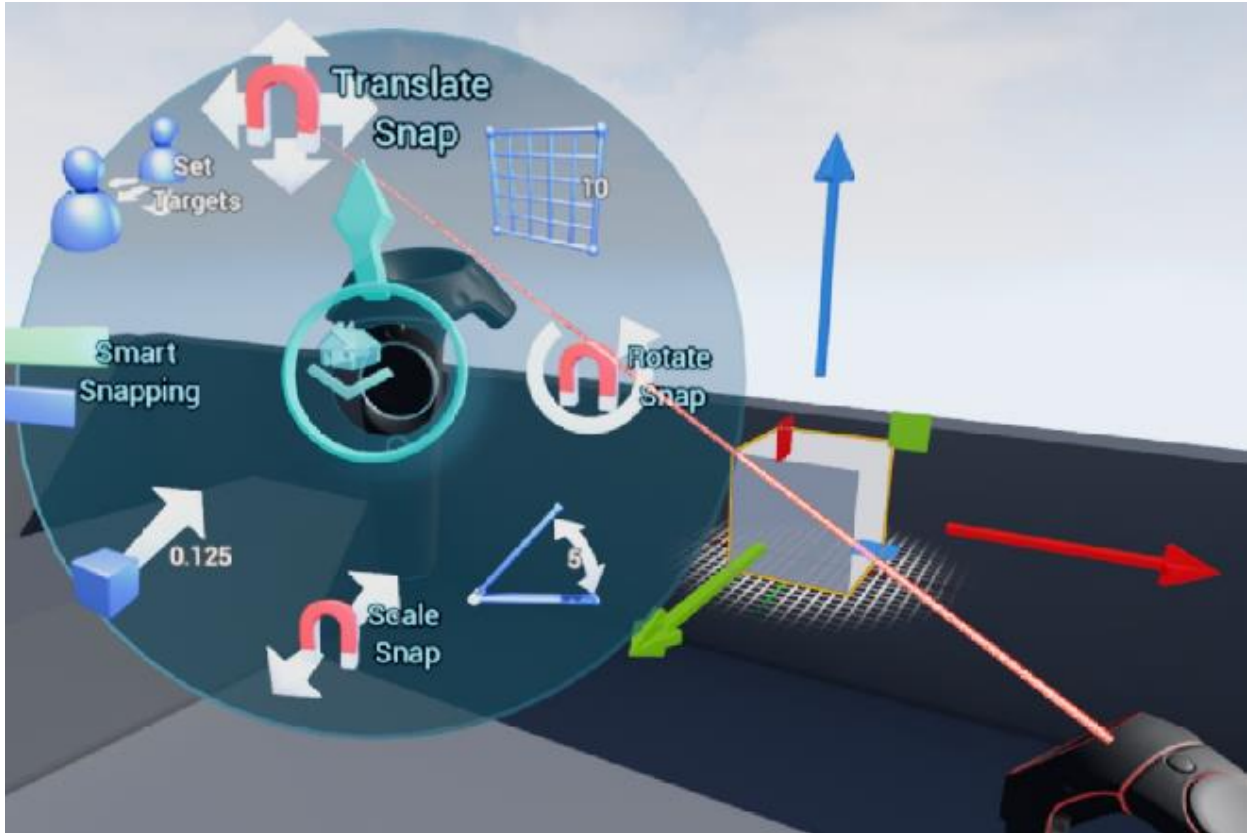
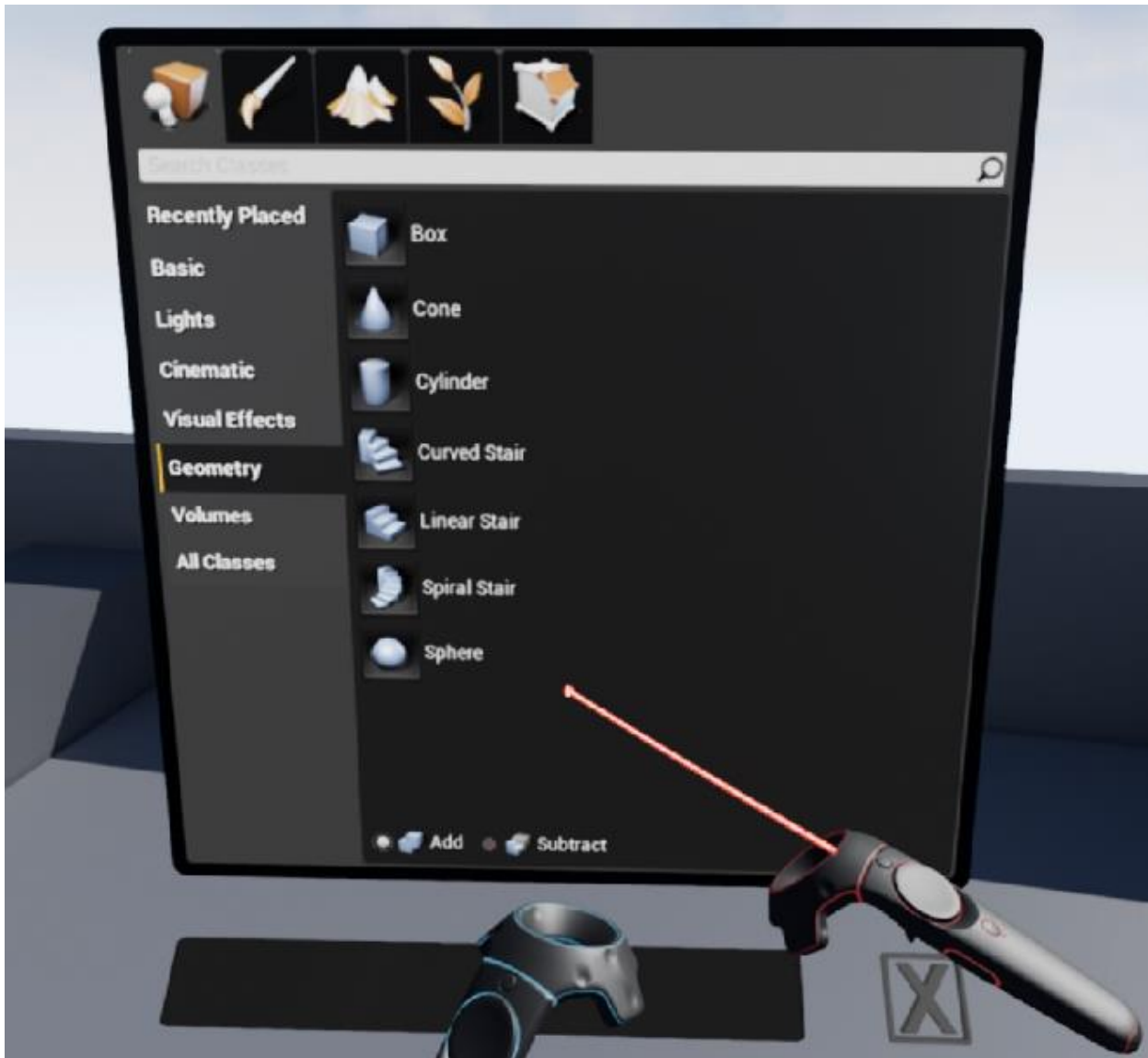


Figure 33 - Unreal: Snapping Menu

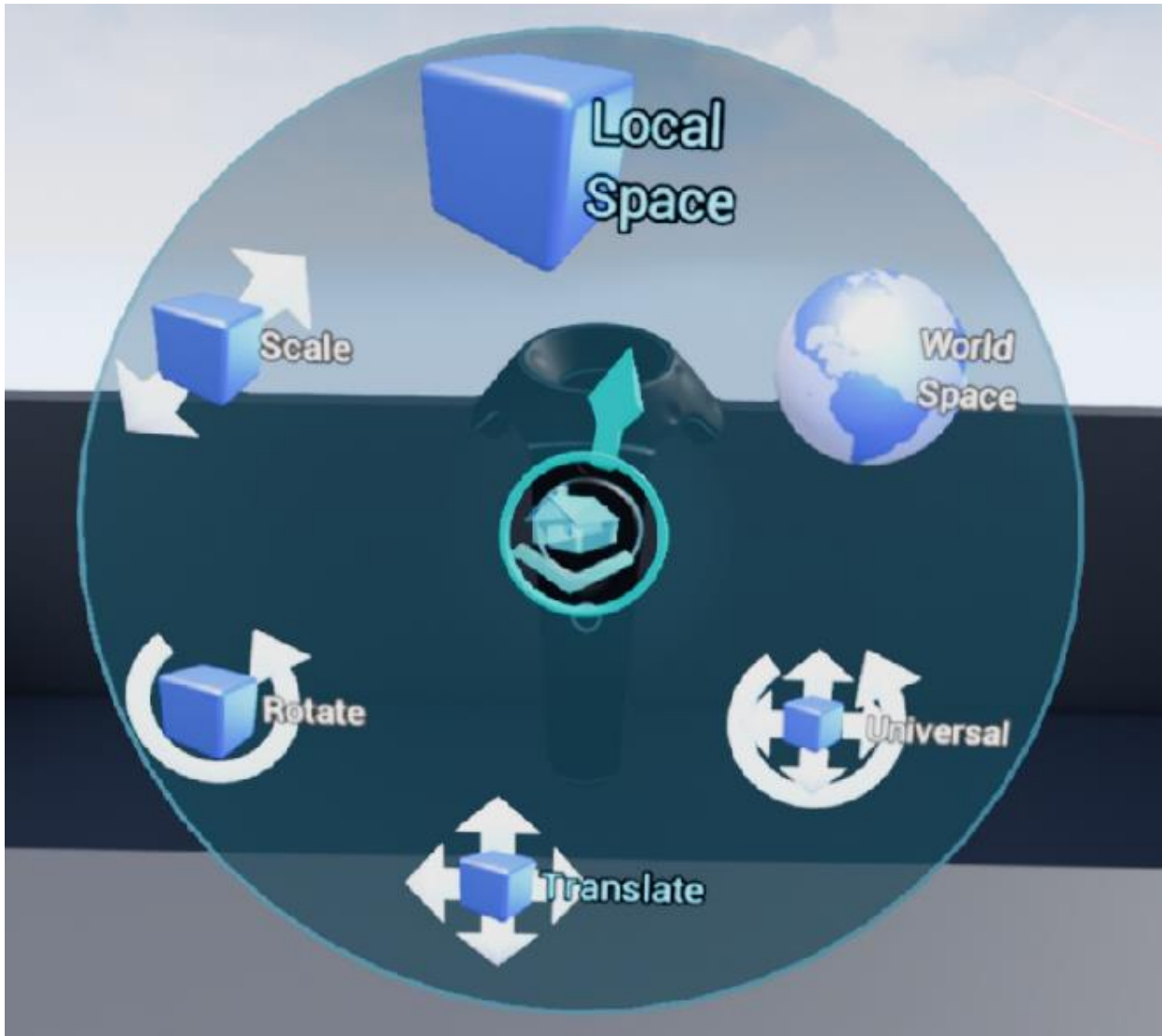
3. Return to the home menu
4. From the “Modes” menu, select “Actors”
5. Choose a primitive to create (figure 34)



*Figure 34 - Unreal: Insert Geometry Menu*



6. Reopen the home menu, select “Gizmo”, and translate, rotate, and scale the primitive (figure 35)



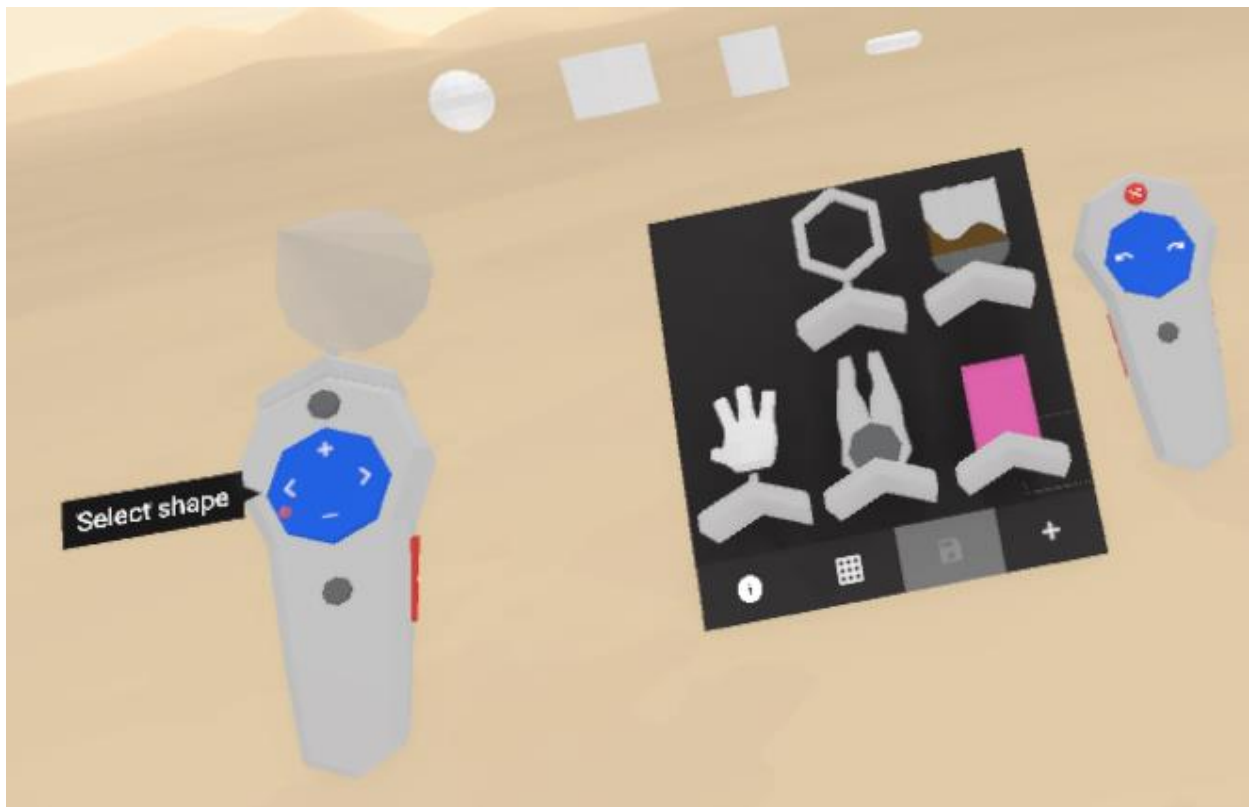
*Figure 35 - Unreal: Gizmo Menu (Translate, Rotate, and Scale)*

7. Repeat steps 4-6 until the model is complete



Google Blocks allows users to create 3D models in VR using primitives (cone, sphere, cube, cylinder, and toroid – see figure 36). It has a grid system that can be toggled on and off to increase precision of translation, rotation, and scaling. Workflow for creating models in Google Blocks was accomplished using these steps:

1. Toggle grid snapping on
2. Select a shape from the shapes tool menu (figure 36)



*Figure 36 - Google Blocks: Shapes Tool*

3. Use the right-hand motion controller to size the shape creation tool and create the shape by holding trigger and dragging the controller through space to create a shape with the desired shape and size (figure 37)



*Figure 37 - Google Blocks: Shape Creation*

4. Select the hand tool
5. Translate and rotate the shape to the desired position and orientation
6. Repeat steps 2-5 until the model is complete

MakeVR allows users to create 3D models using primitives in VR. It has the ability to export models created within the software to a 3D-printable file format. Several months after the initial

release of MakeVR, MakeVR Pro was released, which incorporated the features of grid snapping, mirroring, and measuring using a ruler tool. The Pro version was used for testing in this project.

Model creation workflow for MakeVR Pro was done using the following steps:

1. From the “Files” menu, select “Worlds” and choose a world to work in (figure 38)

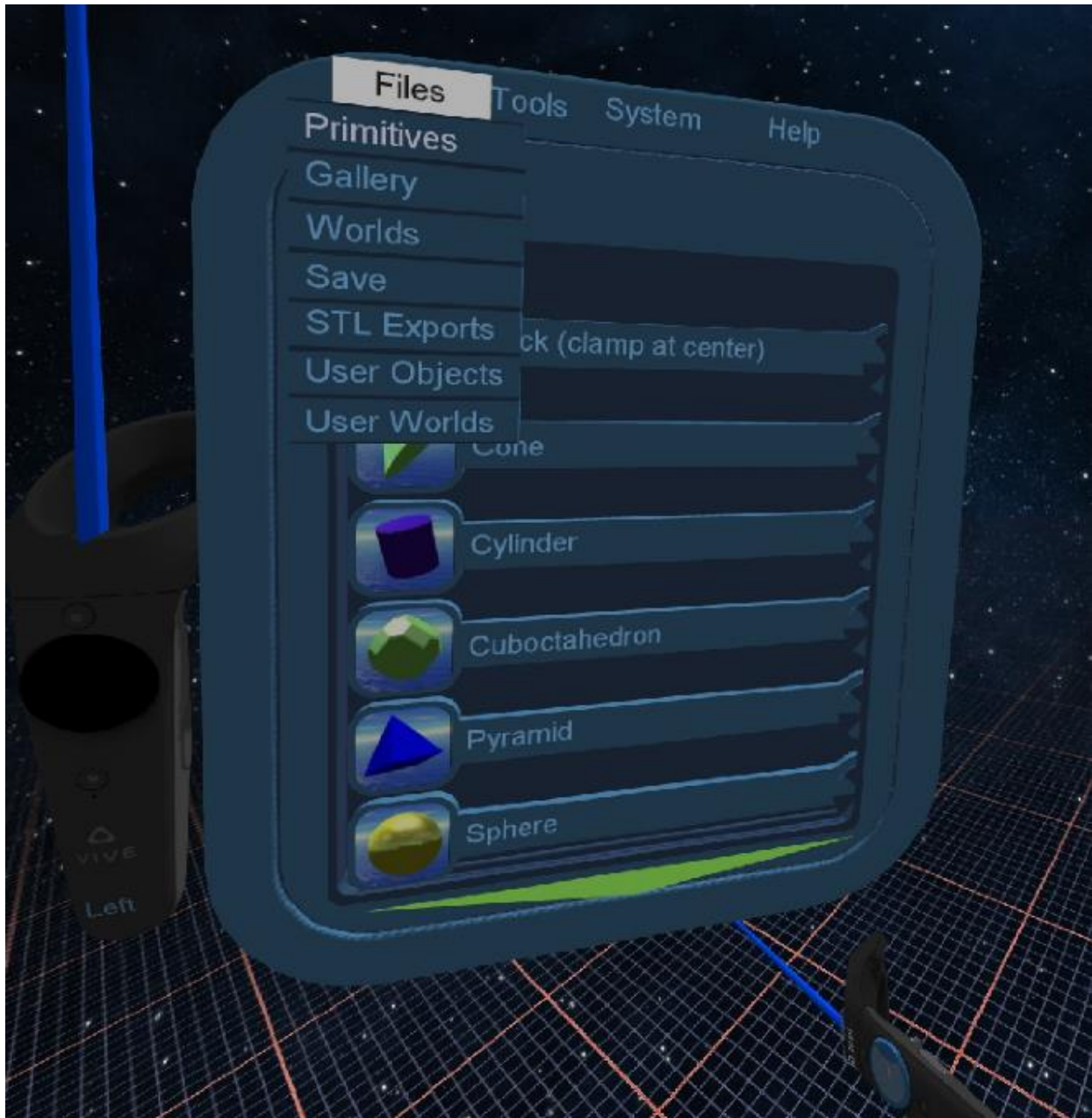


Figure 38 - MakeVR Pro: Files Menu

2. From the “Tools” menu, select the “Grid” tool to make a grid appear (figure 39)



*Figure 39 - MakeVR Pro: Tools Menu*

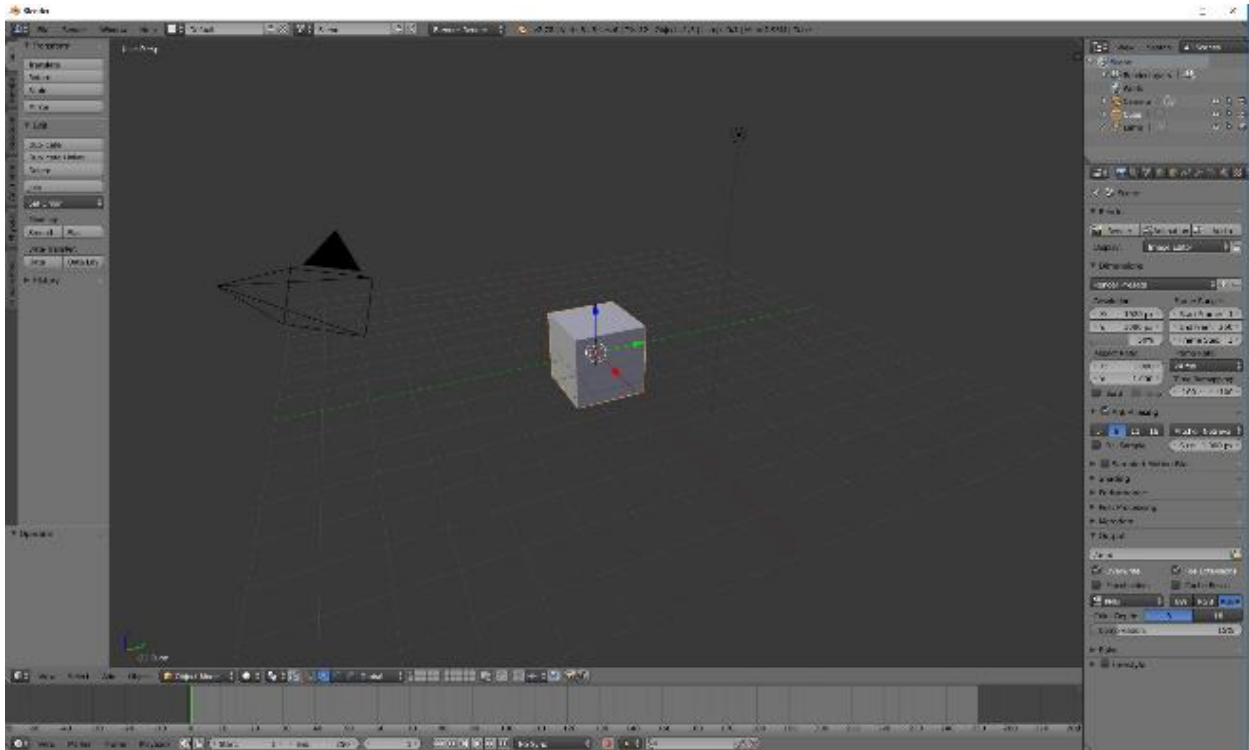
3. Scale the grid to the desired size by grabbing it using the trigger button with both controllers and moving the controllers apart or together
4. Select the “Ruler” tool to make a ruler appear (figure 39)

5. Scale the ruler
6. Select “Primitives” from the “Files” menu and choose a primitive to create (figure 38)
7. Snap the primitive to the grid and/or ruler in the desired orientation by dragging them together
8. From the “Tools” menu, use the sizing tool to change the dimensions of the primitive (figure 39)
9. Resize the primitive by grabbing it with both controllers and moving the controllers apart or together
10. Use the “Add” tool to join primitives together (figure 39)
11. Repeat steps 6-10 until the model is complete
12. Remove the model from the grid and/or ruler by grabbing each and moving them quickly apart

Three traditional interface CAD programs were chosen for this project in order to make comparisons with the VR CAD programs: Blender, FreeCAD, and SolidWorks. These particular programs were chosen because they are widely used and represent three primary use demographics; all three are capable of producing basic 3D models despite having the complex functionality of full-featured CAD programs.

Blender is a full-featured free 3D-modeling program primarily used for video game model creation and animation, but it can also be used to create CAD models. The workflow for creating models in Blender involves the following steps:

1. Blender begins with a cube primitive in the center of the workspace – if the cube is required, translate, rotate, and scale it to the appropriate location, orientation, and size. If the cube is not needed, press delete (figure 40)



*Figure 40 - Blender Home*

2. Click the “Create” menu
3. Select the desired primitive to create
4. Click the “Tools” menu
5. Translate, rotate, and scale the primitive using the 3D view editor at the bottom
6. Repeat steps 2-5 until the model is complete



FreeCAD is a highly recommended free CAD program widely used to create models for 3D-printing. The following steps were used in the workflow for FreeCAD to create models:

1. Click the “Create a new empty document” button
2. Select “Part” from the drop-down menu (figure 41)
3. Click the desired primitives button or click the “Advanced utility to create shapes” button to create 2d shapes and extrude using the “Extrude a sketch” button
4. Translate, rotate, and resize the 3D shape by entering the appropriate numbers into the placement window
5. Repeat steps 3-4 until the model is complete

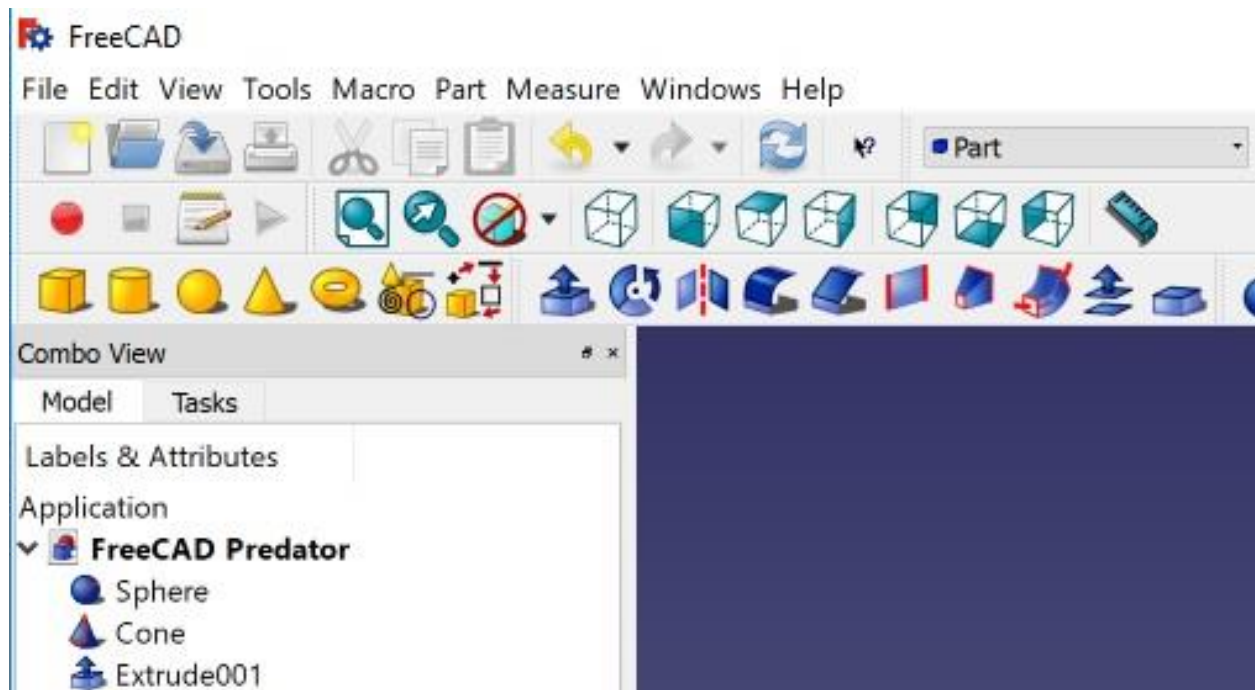


Figure 41 - FreeCAD "Part" Screen

SolidWorks is one of the most full-featured CAD programs available today and widely used in industry to design products of all types. Workflow for SolidWorks was performed using the following procedure:

1. Click the “New” button
2. Select “Part” from the window and click “OK”
3. Click the “Sketch” tab (figure 42)



Figure 42 - SolidWorks: Sketch Tab

4. Select the appropriate type of sketch needed to extrude to the 3D shape needed
5. Select the plane on which to draw the sketch
6. Draw the sketch and click the checkmark
7. Set constraints for the sketch (dimensions and distances from the origin) by clicking a point or line on the sketch, holding Ctrl, and clicking another point or line on the sketch
8. Enter the required value and click the checkmark



9. Extrude the sketch to the desired length and click the checkmark
10. Right-click the 3D shape and click “Move”
11. Translate and rotate the shape into the desired position
12. Repeat steps 3-11 until the model is complete

Three unique models were chosen to create in each CAD program and in the following order: an airplane, a quad-rotor helicopter (quadcopter), and a Predator drone. Times to create each model in every program were recorded and my impressions of each were noted. There was no prior training or experience with any of the CAD programs except for the one created using Unreal Engine.

## VI. RESULTS

Each of the VR CAD programs took significantly less time to learn how to operate than the traditional CAD programs; however, once familiar with the workflow and interfaces of each, model creation time was greatly reduced (table 1).

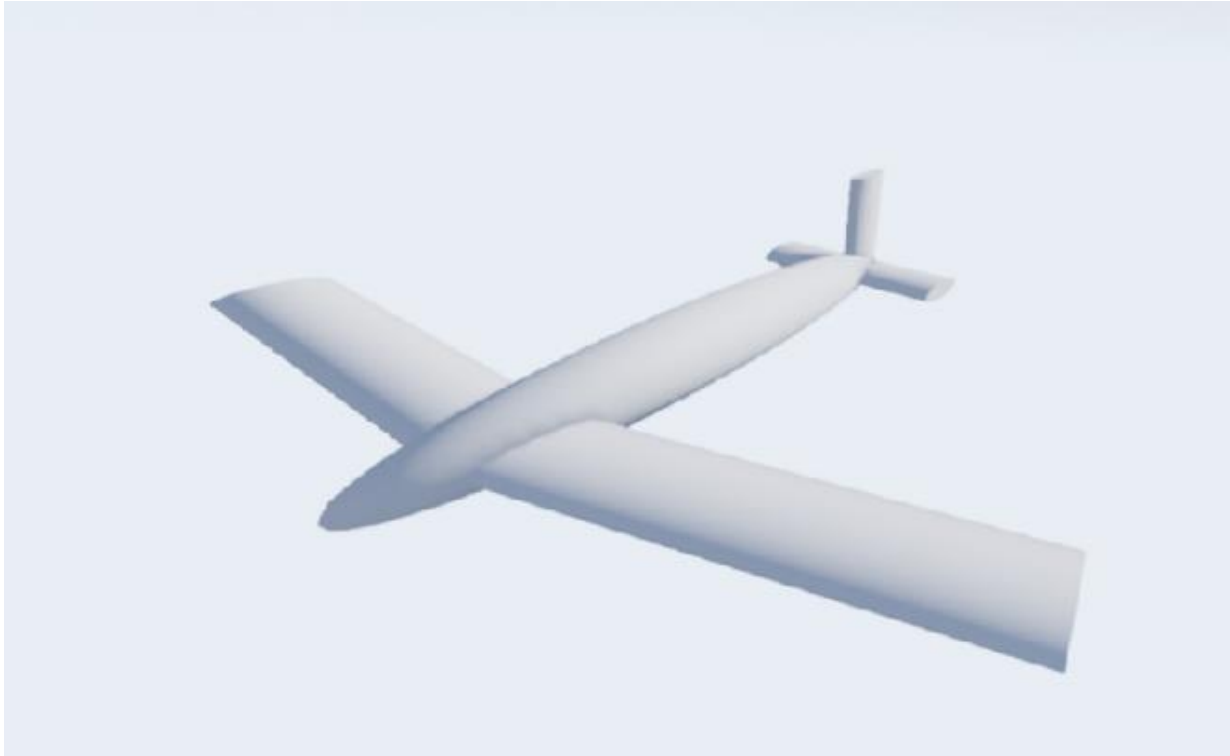
Table 1 - Model Creation Times

		Time (minutes)		
		Airplane	Quadrocopter	Predator
<b>VR CAD</b>	Unreal	30	30	20
	MakeVR Pro	30	20	15
	Google Blocks	19	15	10
<b>Traditional CAD</b>	Blender	60	50	20
	FreeCAD	300	240	60
	SolidWorks	300	45	40

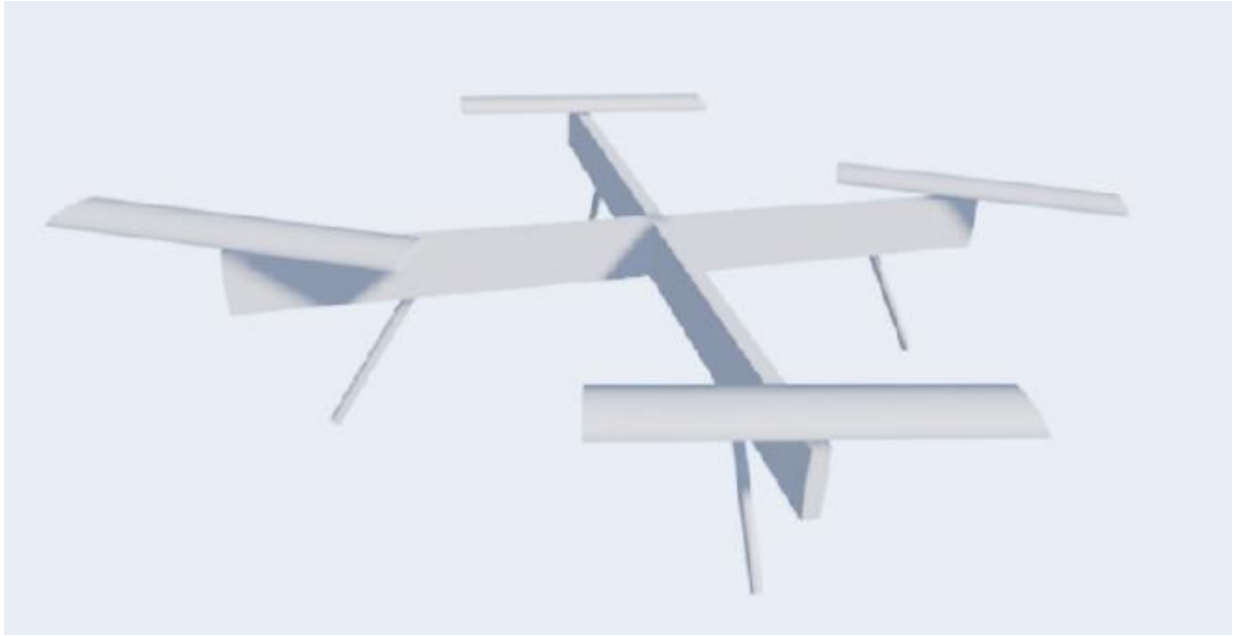
Except for Unreal, each model was created without any prior knowledge of how to use any of the tested CAD programs. Because I was familiar with the Unreal VR CAD interface, the airplane model creation time is not an accurate estimate of time to learn Unreal VR CAD and produce a first model. However, the other times are valid and a good indication of the learning curve for each program based on level of complexity and intuitiveness of each program.

The Unreal VR CAD program was reasonably easy to work with; in order to expedite the development of the software, all of the original Unreal menus were maintained as part of the program. Simplifying the menus would reduce the initial learning curve and model creation time as a whole, but probably not by much as all menu items are within four steps of the home menu. The Unreal software is reasonably stable and bug free but there were some issues worth noting. First, Unreal crashed on two occasions (not as a result of typical model creation workflow

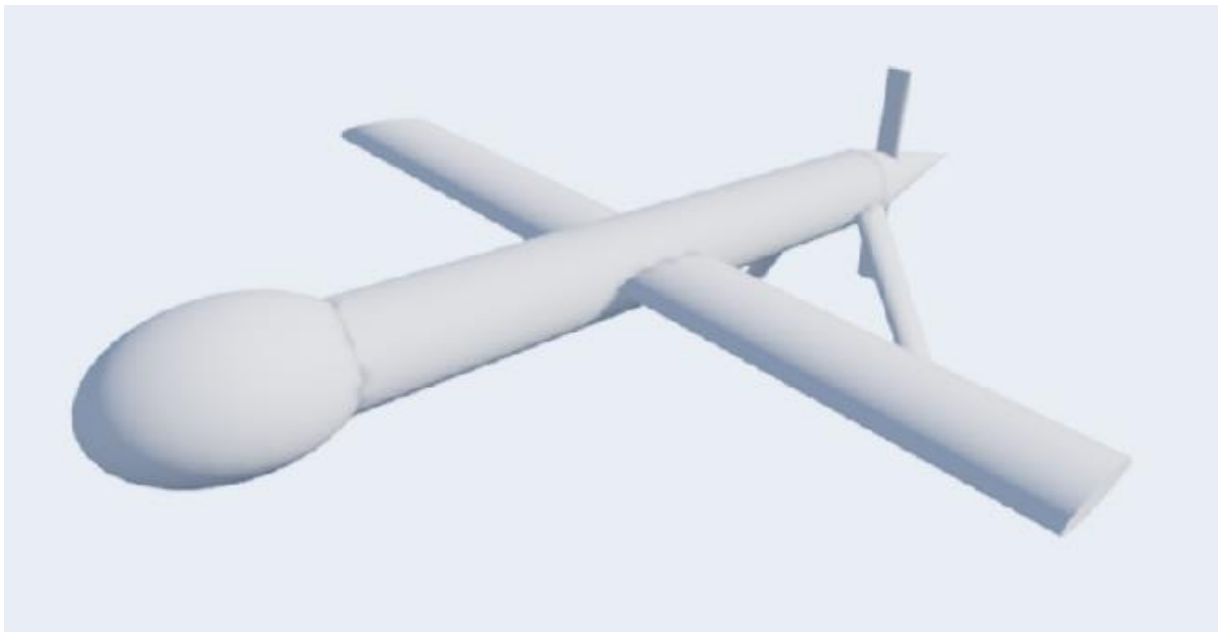
processes), and second, rotation commands would start to behave erratically if rotating along multiple axes.



*Figure 43 - Unreal Airplane*



*Figure 44 - Unreal Quadcopter*



*Figure 45 - Unreal Predator*

MakeVR Pro is a feature-rich and intuitive VR CAD program that uses several unique approaches to building models. There is a tutorial that runs automatically the first time MakeVR Pro is started (users can also rerun it through the “Help” menu – see figure 46) and the “Tools” and “System” menus have a “Show Me” button that briefly illustrates how to use the currently selected tool (figure 39). While not as fully-featured as the traditional interface CAD programs, MakeVR Pro has more capabilities than the other VR CAD programs and is very intuitive to use and easy to learn and produce models. It is the newest of all the CAD programs, having only been released several days before testing, and there were issues with the tutorial series that prevented me from completing two of the modules; they were skippable, which allowed me to proceed past the broken tutorials. Luckily, most features were also demonstrated with the “Show Me” feature mentioned before.

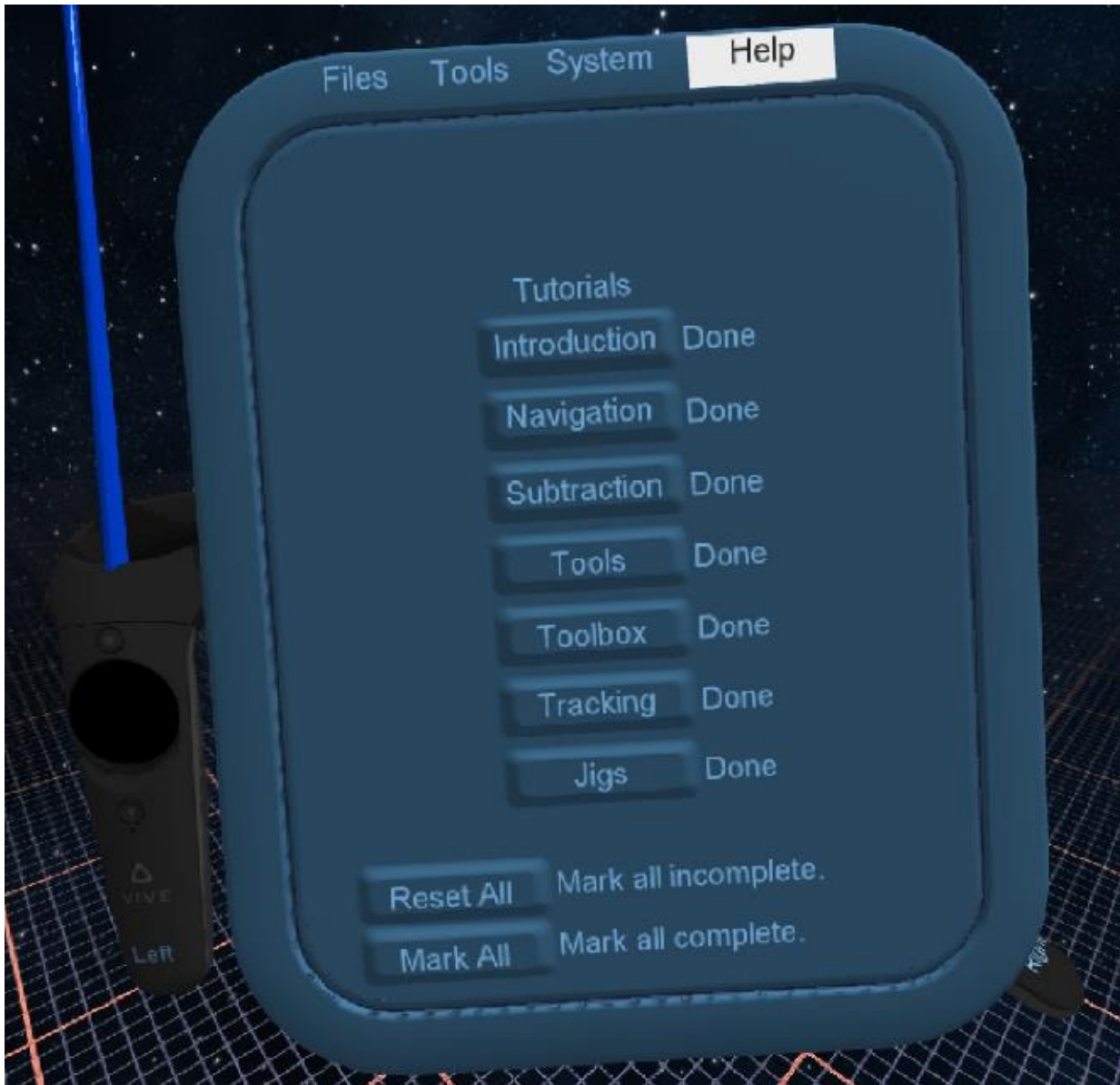
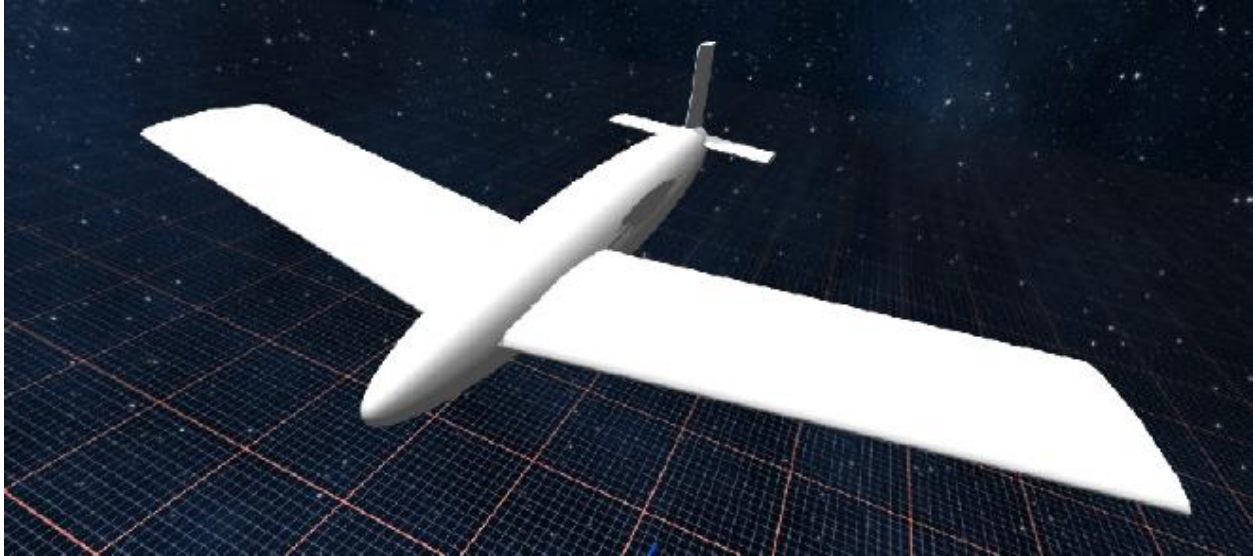
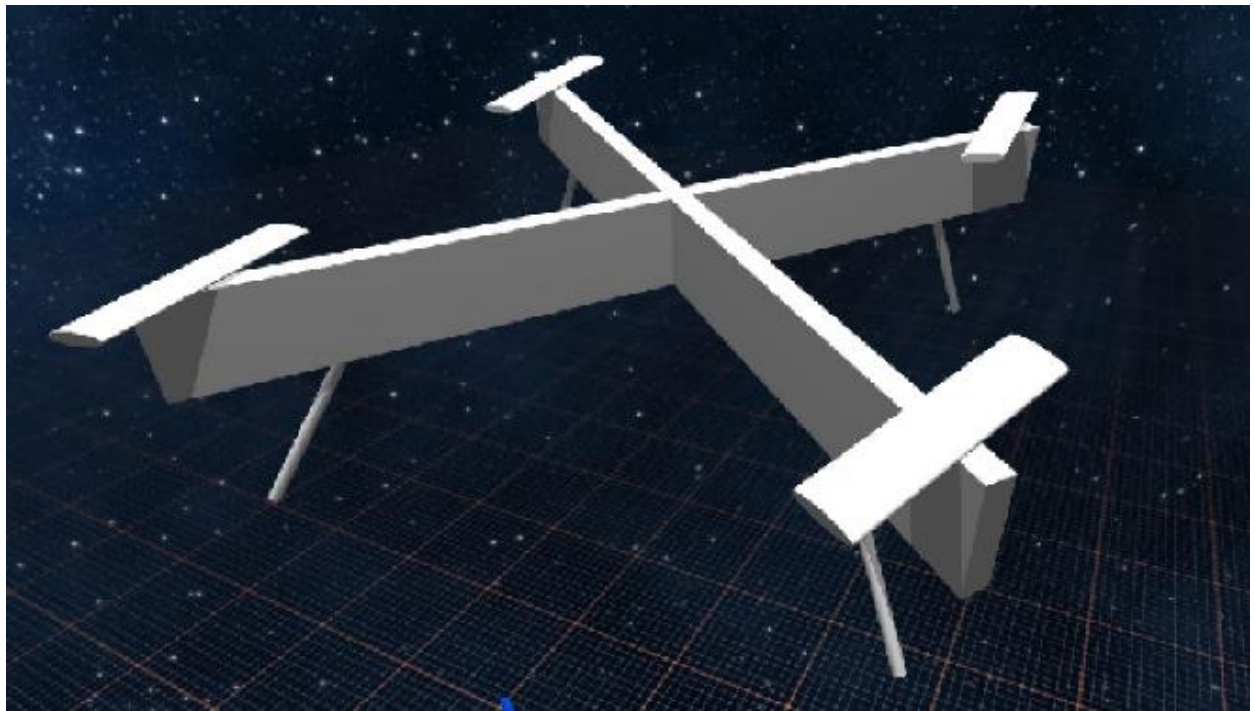


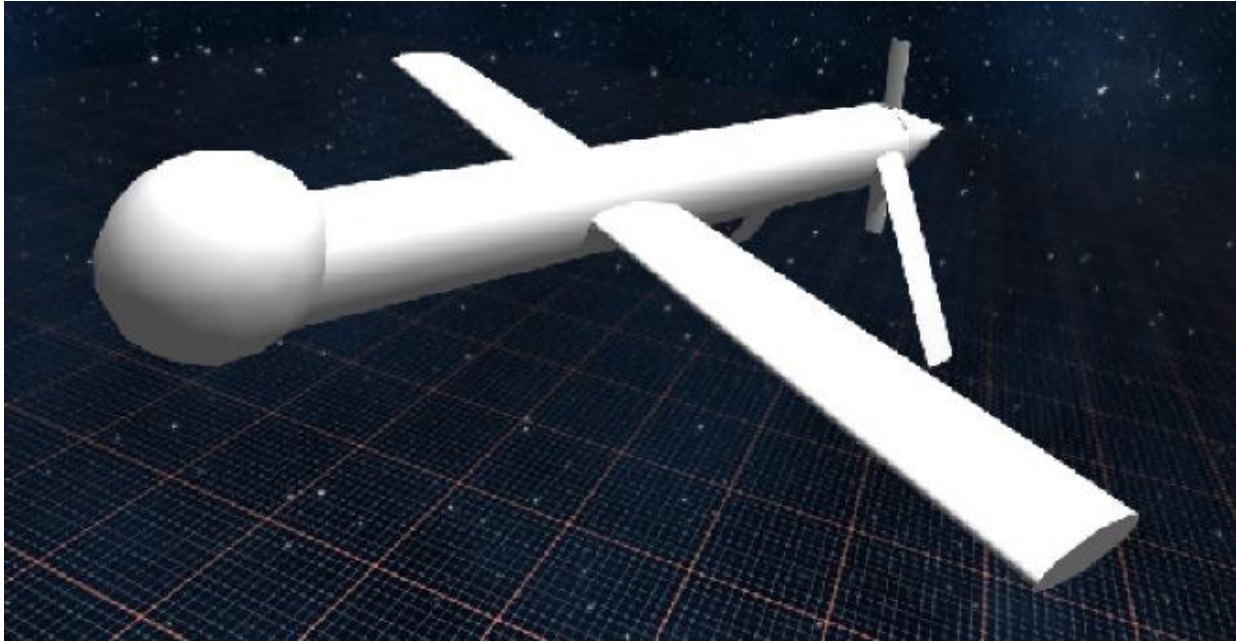
Figure 46 - MakeVR Pro Help Menu



*Figure 47 - MakeVR Pro Airplane*



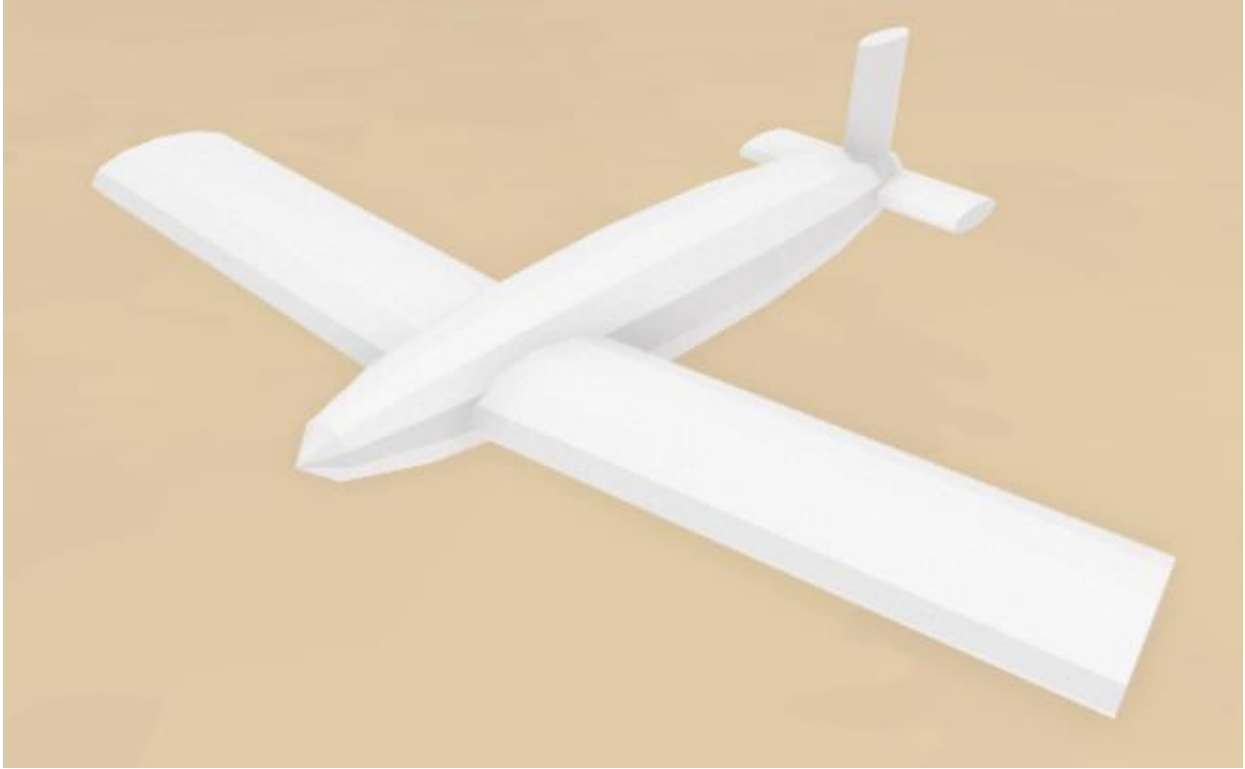
*Figure 48 - MakeVR Pro Quadcopter*



*Figure 49 - MakeVR Pro Predator*

Google Blocks has the least features, but was the most intuitive to use. There is a tutorial upon first running the software, which leads users through basic features, but neglects to show how to use primitives shaping and the grid feature, both of which were necessary for this project. Google Blocks models have low polygon counts by default (more polygons can be added, but only one at a time) and cannot be exported to other file formats. During my use, the software was bug free. One glaring feature omission is the inability to join primitives together into a single model. To move a model, users must select each shape in the model individually by moving the motion controller through each shape with the trigger pulled. Only then can the assemblage of primitives be translated, rotated, and scaled as a whole.

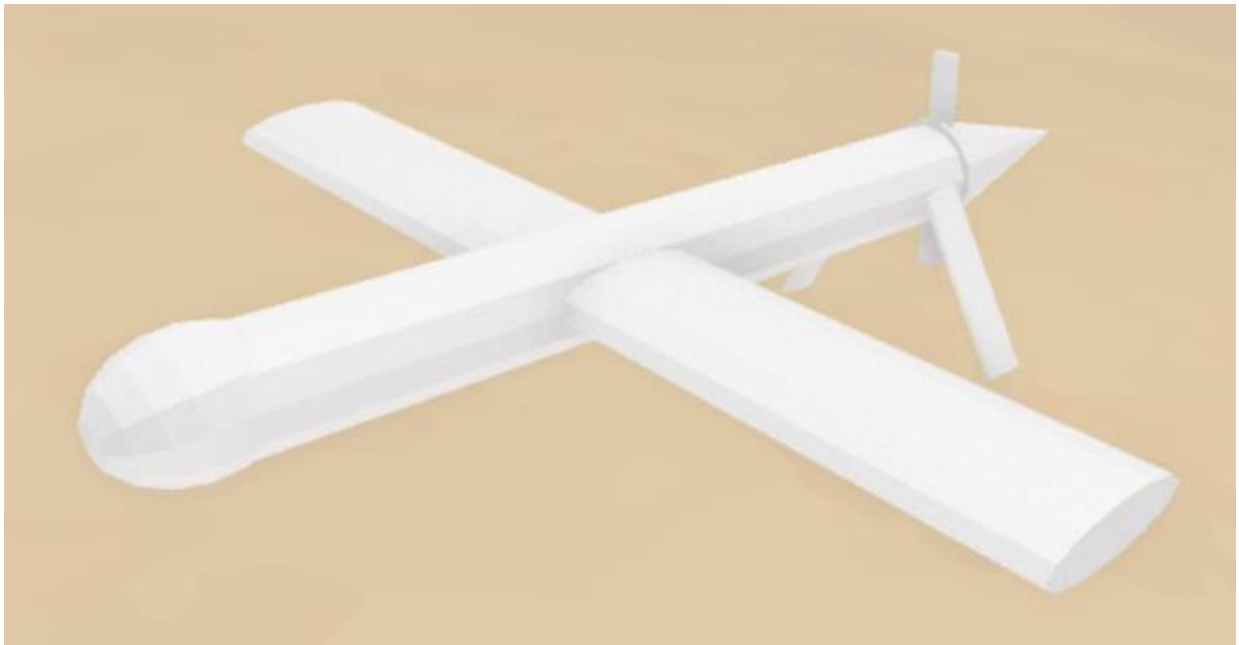




*Figure 50 - Google Blocks Airplane*

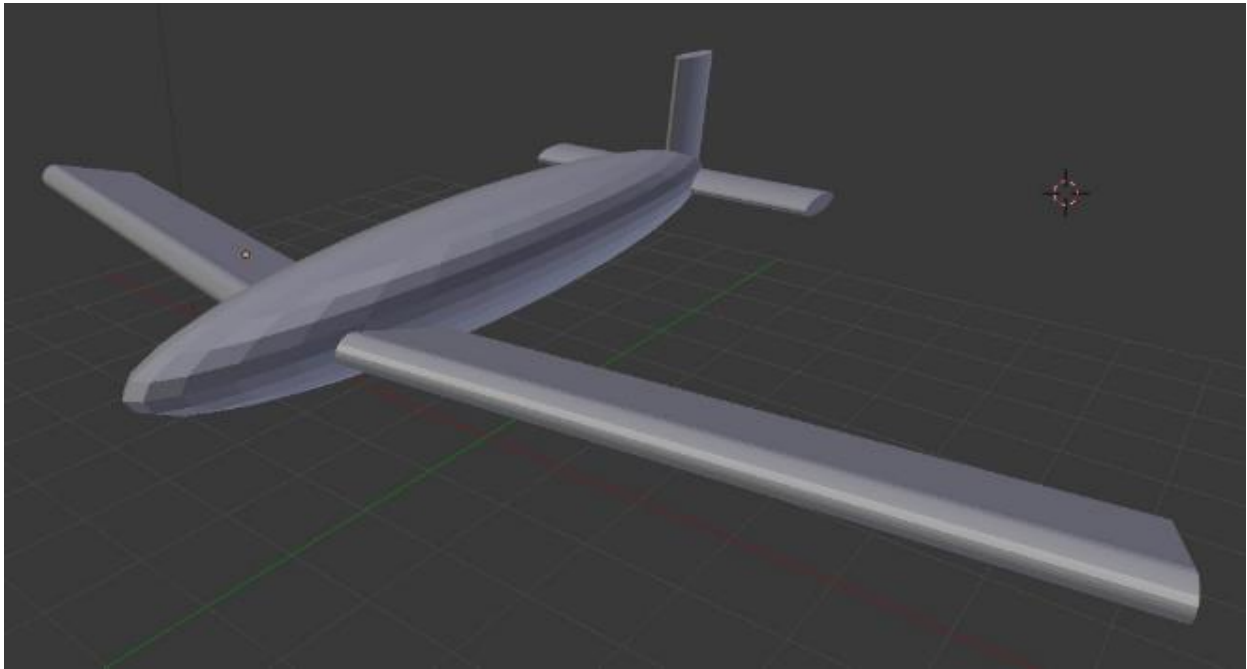


*Figure 51 - Google Blocks Quadcopter*

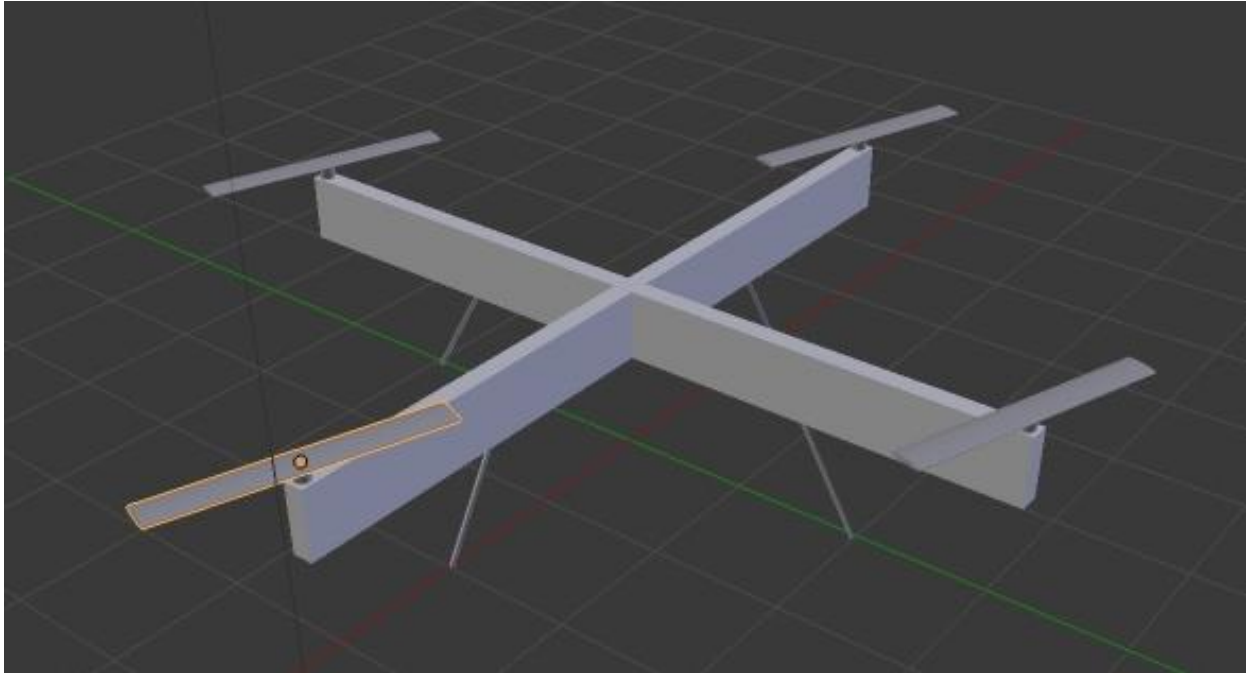


*Figure 52 - Google Blocks Predator*

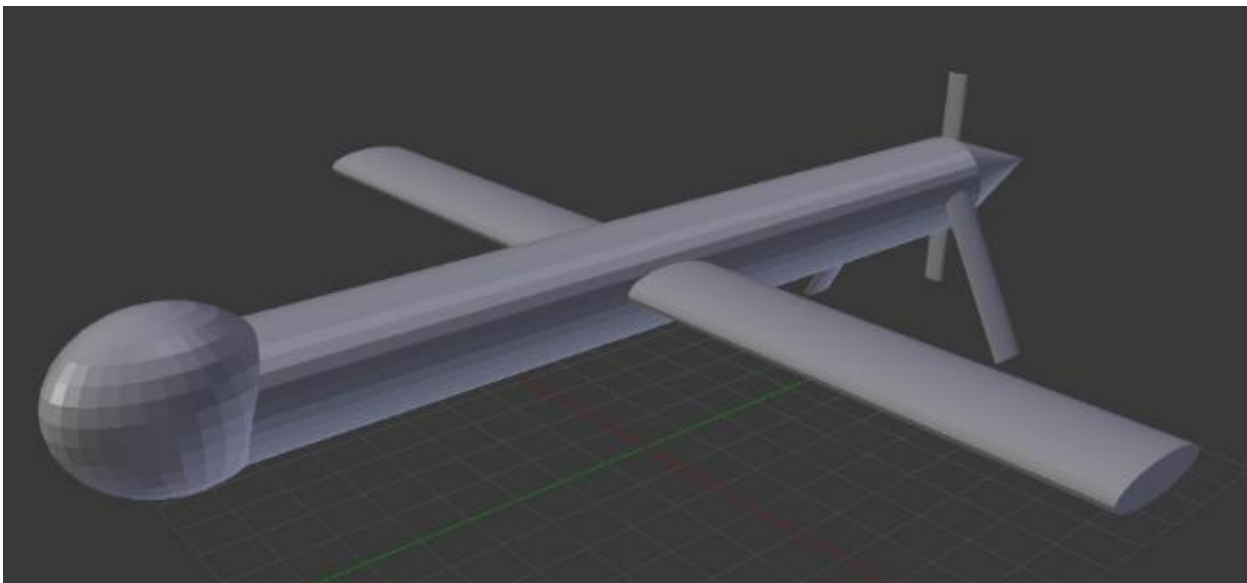
Blender can be used for many applications including animation and CAD model creation. Models can be as precise as CAD, with all dimensions able to be specified or unspecified. Though it is complex with many features, it is fairly intuitive to use even for users with no prior CAD experience. No bugs were experienced while using this software.



*Figure 53 - Blender Airplane*



*Figure 54 - Blender Quadcopter*



*Figure 55 - Blender Predator*

FreeCAD was the least intuitive and buggiest of all the CAD programs tested. The start page has a tiny “x” at the bottom that users must click before they can begin using the software (figure 56) and the process of learning it was long and arduous. Translation, rotation, and scaling can only be done by entering the desired values into a menu and subsequently entered values cause shapes to relocate unpredictably, particularly when entering rotation values. The software crashed several times during model creation and primitives would relocate so unexpectedly that it was easier to revert to previous saves than try to fix the model from the current state because the undo feature was also unreliable.

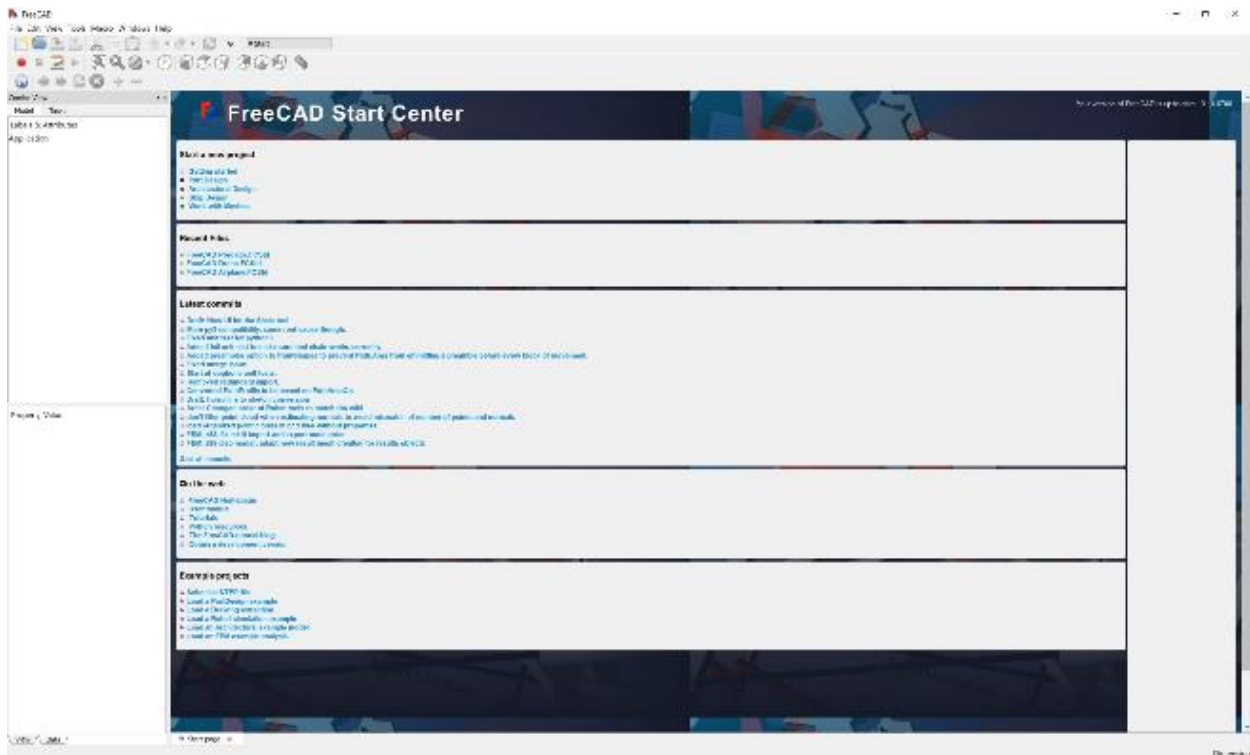
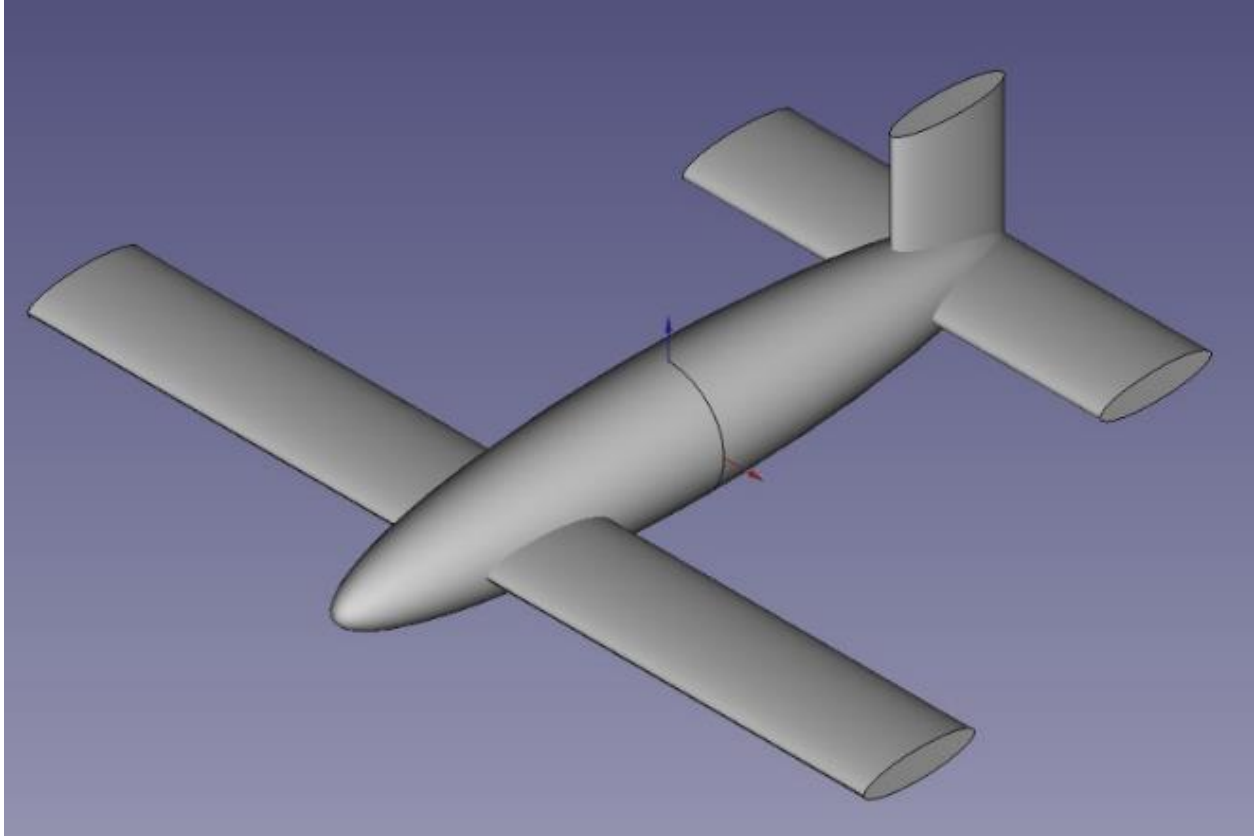
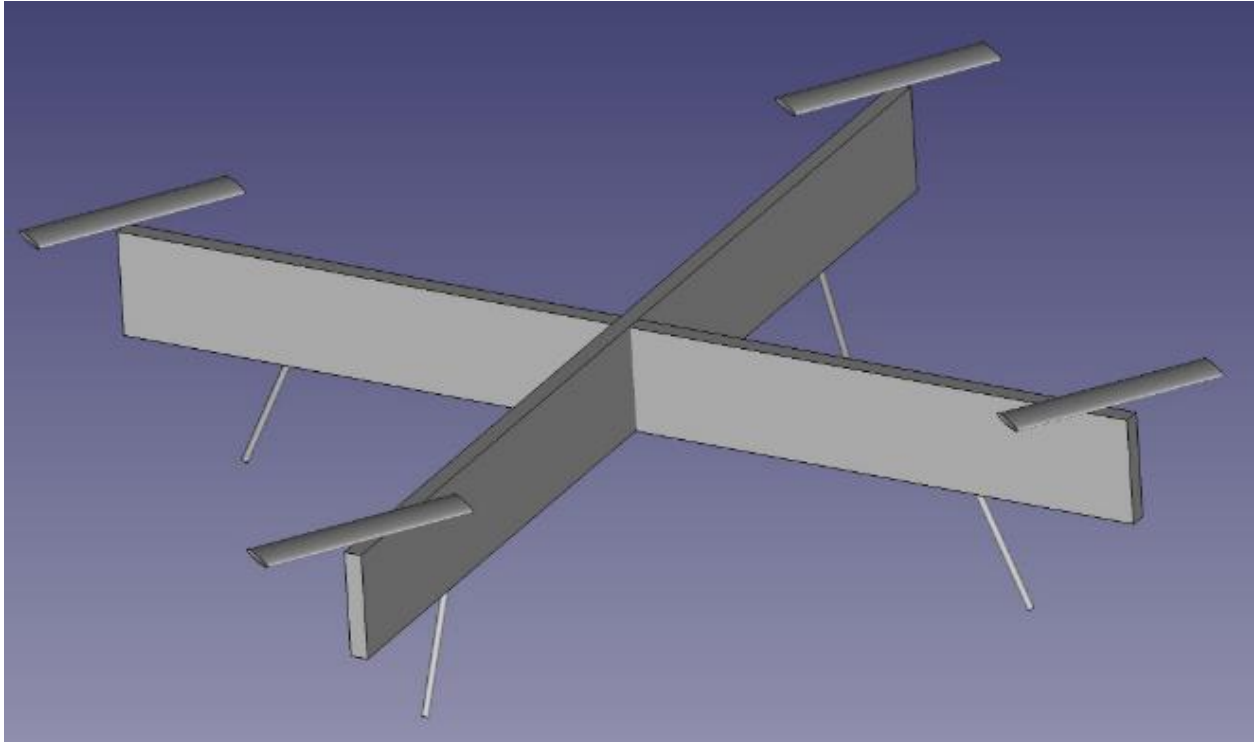


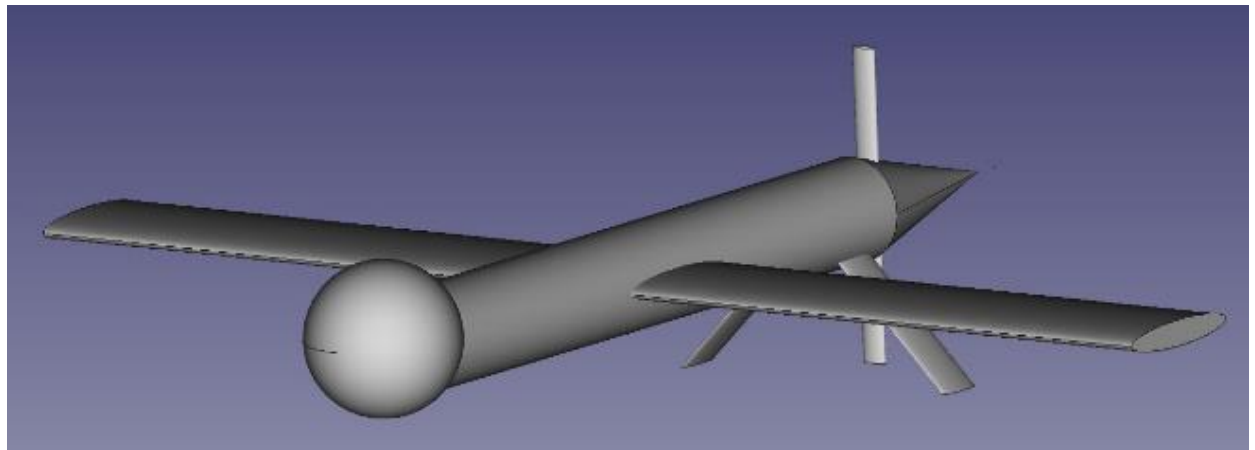
Figure 56 - FreeCAD Start Page



*Figure 57 - FreeCAD Airplane*



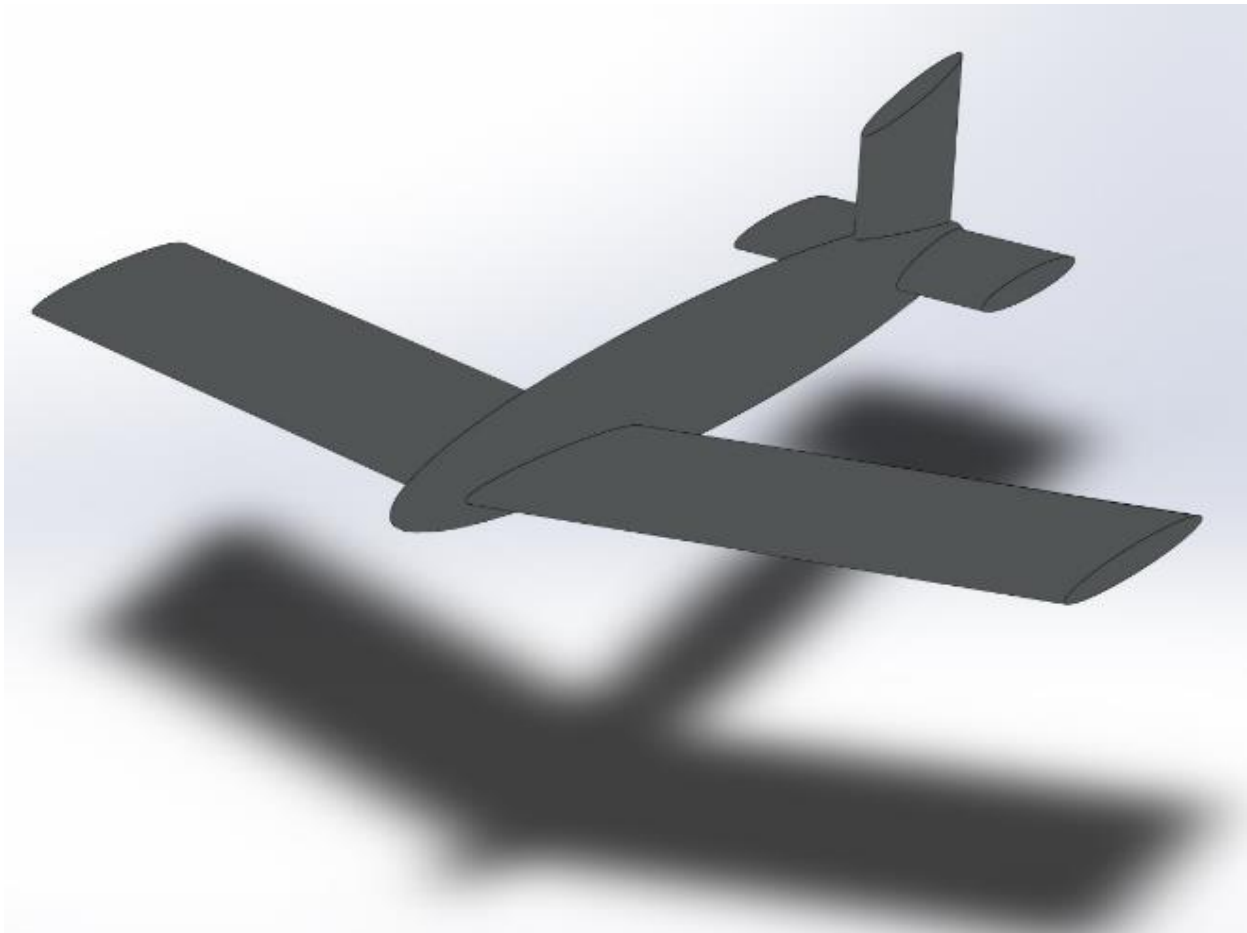
*Figure 58 - FreeCAD Quadcopter*



*Figure 59 - FreeCAD Predator*

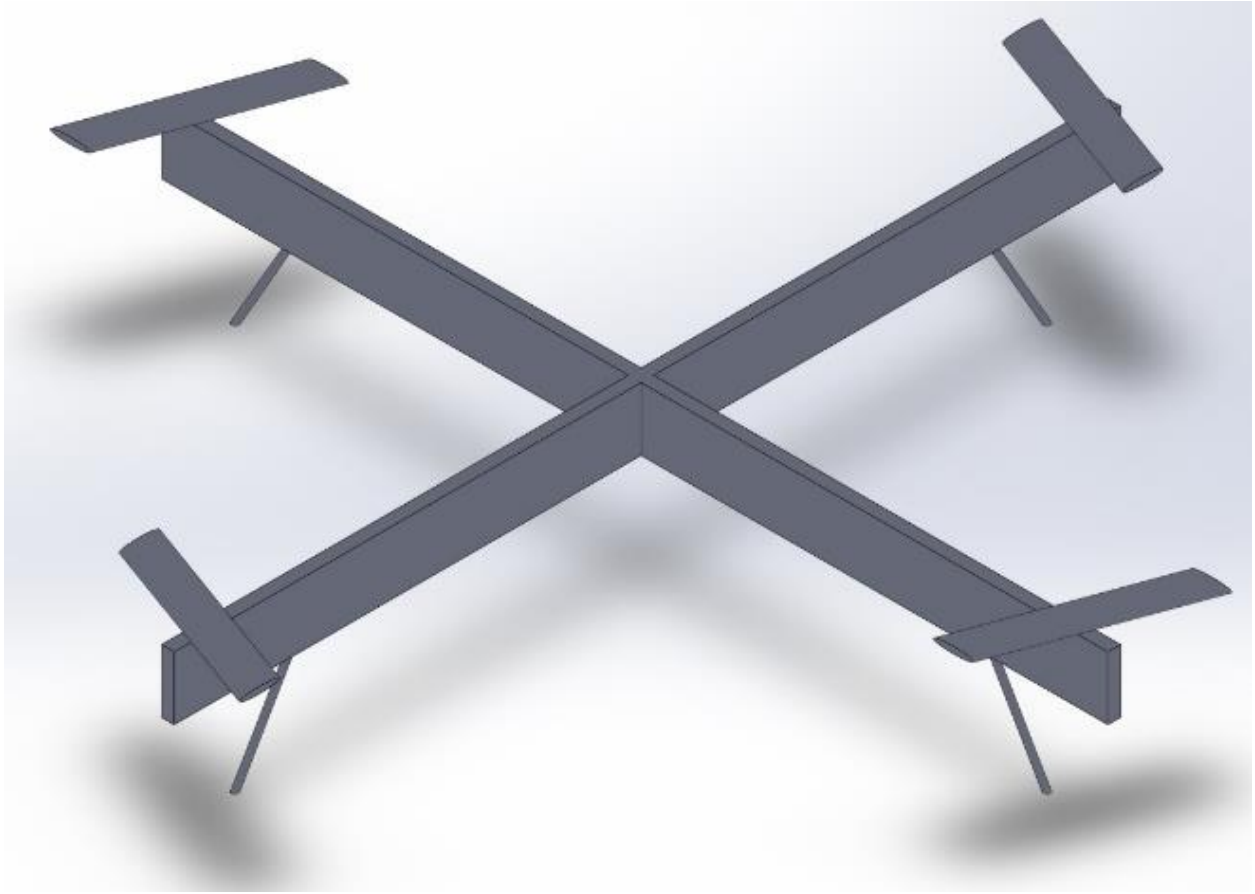
SolidWorks was the fullest-featured program tested and what many would consider archetypal for CAD software since it is widely used and shares many features in common with other commonly

used CAD programs. The learning curve was relatively steep for this program, but once learned, workflow was logical, and timely model creation became possible. During testing, there was a bug where the display window would create a quarter-sized version of itself in the upper-right corner of the screen, but that may have been an issue with the installation rather than the software itself. The installation of SolidWorks was a very complicated process with lots of installers, sub-installers, product verification, and digital rights management involved.

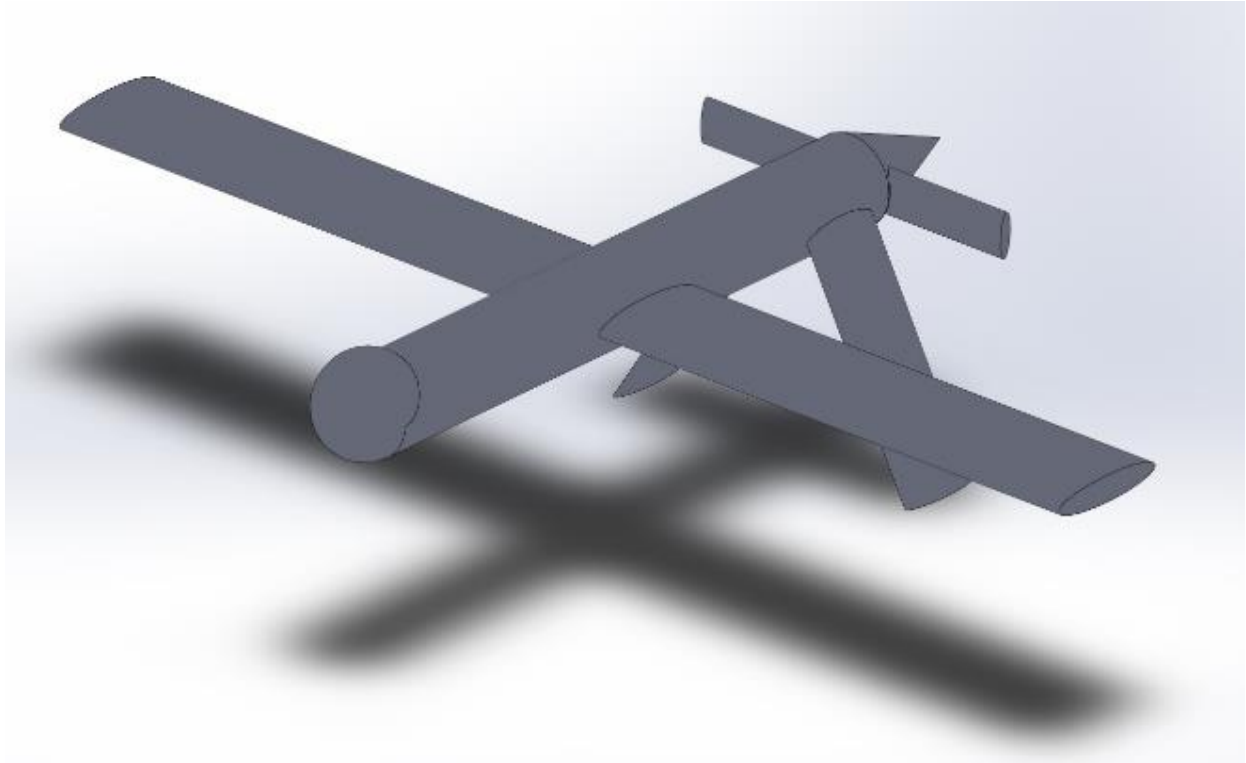


*Figure 60 - SolidWorks Airplane*





*Figure 61 - SolidWorks Quadcopter*



*Figure 62 - SolidWorks Predator*

## VII. CONCLUSION

VR CAD programs tested for this project were by far easier to learn, more intuitive to use, and faster to produce models both prior to and after becoming familiar with each program than the traditional interface CAD programs. At this time, each of the traditional interface CAD programs were more feature-rich than any of the VR CAD programs, but there is no reason that this should continue to be true. Even for CAD with similar types and amounts of features, VR manipulation of 3D objects is clearly more intuitive than through a 2D interface because every person is accustomed to manipulating physical objects in 3D space.

Disadvantages of VR, as it is today, are that it can be fatiguing if done for more than several hours at a time; the headset is somewhat heavy and the eyes are forced to focus on a screen close to the face. I found fatigue to be more of an issue while standing, however, model creation can be done sitting down or standing up, so many users may consider that option an advantage over traditional interface CAD programs. Also, taking a short break every two or three hours alleviated any fatigue I was experiencing at the time.

## VIII. FUTURE WORK

It seems certain as time progresses that more fully-featured VR CAD programs will be developed. During the course of several months of this project, two VR CAD programs were released: Google Blocks and MakeVR and MakeVR released a Pro version with the ability to measure and snap within two months after their initial release. It was my intent to advocate with this paper that using VR for CAD would be a good use of the technology, but it seems the word is already out. It is my hope that the current major developers of traditional interface CAD will begin to take their knowledge, experience, and expertise and apply it to creating CAD programs in VR.

There are also plenty of improvements to be made in the area of VR technology. The headset could benefit greatly from being lighter and less bulky with higher resolution, eye tracking, integrated sound, and haptic feedback. Controllers could benefit from being lighter and less bulky with better haptic feedback and a more thoroughly tested button layout. Many people favor eliminating controllers entirely – using gloves or hands only and tracking hand motion and orientation with cameras and/or other sensors. Moving around in VR is also a point of concern;

there are multidirectional treadmills and virtual teleportation systems, but the problem is far from solved. It seems it may be a long road to the holodeck but fortunately, it looks as if there is no shortage of people willing to pave the way.

## REFERENCES

- [1] Alvarez, Juan C., and Hai-Jun Su. "VRMDS: An Intuitive Virtual Environment for Supporting the Conceptual Design of Mechanisms." *Virtual Reality*, vol. 16, no. 1, 2012., pp. 57-68doi:10.1007/s10055-009-0144-z.
- [2] Berta. "Integrating VR and CAD." *IEEE Computer Graphics and Applications*, September/October, 1999, pp. 14-19.
- [3] Bourdot, Convard, et al. "VR-CAD integration: Multimodal immersive interaction and advanced haptic paradigms for implicit edition of CAD models." *Computer-Aided Design*, vol. 42, 2010, pp. 445-461.
- [4] Eshaq, Karboulonis. "Design consideration for the design of an advanced VR interface for knowledge management and its relevance to CAD." *Automation in Construction*, vol. 12, 2003, pp. 501-507.
- [5] Fiorentino, Uva, et al. "Improving bi-manual 3D input in CAD modelling by part rotation optimisation." *Computer-Aided Design*, vol. 42, 2010, pp. 462-470.
- [6] Gîrbacia, Florin. *Evaluation of CAD Model Manipulation in Desktop and Multimodal Immersive Interface*, vol. 327, Trans Tech Publications Ltd, Zurich, 2013.doi:10.4028/[www.scientific.net/AMM.325-326.289](http://www.scientific.net/AMM.325-326.289).
- [7] Huang, Jinmiao. *Intuitive Interface for Conceptual 3D Modeling*, ProQuest Dissertations Publishing, 2015.
- [8] Kang, Moon, et al. "Application of VR Technology linked with 4D CAD System for Construction Project." *2009 International Conference on New Trends in Information and Service Science*, 2009, pp. 1058-1062doi:10.1109/NISS.2009.139.

- [9] Kosmadoudi, Z., et al. "Engineering Design using Game-Enhanced CAD: The Potential to Augment the User Experience with Game Elements." *COMPUTER-AIDED DESIGN*, vol. 45, no. 3, 2013., pp. 777-795doi:10.1016/j.cad.2012.08.001.
- [10] Kostic, Zona, et al. "Comparative Study of CAD Software, Web3D Technologies and Existing Solutions to Support Distance-Learning Students of Engineering Profile." *International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 4, 2012., pp. 181-187.
- [11] Leu, Ming C., et al. "CAD Model Based Virtual Assembly Simulation, Planning and Training." *CIRP Annals - Manufacturing Technology*, vol. 62, no. 2, 2013., pp. 799-822doi:10.1016/j.cirp.2013.05.005.
- [12] Liu, Chao. *Using Virtual Reality to Improve Design Communication*, ProQuest Dissertations Publishing, 2012.
- [13] Luo, Li, Liu, and Li. "Key Technology Research on Data Conversion Between B-Rep Based CAD and VR Model." *2009 International Conference on Measuring Technology and Mechatronics Automation*, 2009, pp. 104-107doi:10.1109/ICMTMA.2009.606.
- [14] MacAllister, Anastacia M. *Natural User Interfaces for Interdisciplinary Design Review using the Microsoft Kinect*, ProQuest Dissertations Publishing, 2015.
- [15] Maleshkov, Stoyan, and Dimo Chotrov. "Post-Processing of Engineering Analysis Results for Visualization in VR System." *International Journal of Computer Science Issues (IJCSI)*, vol. 10, no. 2 Part 2, 2013., pp. 258.
- [16] Noon, Christian, et al. "A System for Rapid Creation and Assessment of Conceptual Large Vehicle Designs using Immersive Virtual Reality." *Computers in Industry*, vol. 63, no. 5, 2012., pp. 500-512doi:10.1016/j.compind.2012.02.003.

- [17] Pavlik, Ryan A. *Enabling Natural Interaction for Virtual Reality*, ProQuest Dissertations Publishing, 2014.
- [18] Prado, et al. "A Methodology for Reducing the Time Necessary to Generate Virtual Electric Substations." *IEEE Virtual Reality Conference 2016*, March 19-23, 2016, pp. 163-164.
- [19] Pratama, Lucky A. *Immersive Virtual Reality Prototype for Evaluating 4D CAD Model*, ProQuest Dissertations Publishing, 2015.
- [20] Rabätje. "Integration of Basic CAD Functions into a VR Environment." *Volkswagen AG VRLab*.
- [21] Raposo, Soares, et al. "Environ: Integrating VR and CAD in Engineering Projects." *IEEE Computer Graphics and Applications*, 2009, pp. 91-95.
- [22] Si-Qun, Zhi-Li. "A Solution to Integrate CAD/CAE Systems and VR for Complex Product Design Processes." *Second International Conference on Information and Computing Science*, 2009, pp. 331-333doi:10.1109/ICIC.2009.394.
- [23] Stark, Israel, and Wöhler. "Towards hybrid modelling environments – Merging desktop-CAD and virtual reality-technologies." *CIRP Annals – Manufacturing Technology*, vol. 59, 2010, pp. 179-182.
- [24] "SteamVR Development." Unreal Engine. N.p., n.d. Web. 25 Aug. 2016.  
<<https://docs.unrealengine.com/latest/INT/Platforms/SteamVR/index.html>>
- [25] Sung, Ritchie, et al. "Automated design process modelling and analysis using immersive virtual reality." *Computer-Aided Design*, vol. 41, 2009, pp. 1082-1094.
- [26] Thompson, Frederick V., III. *Evaluation of a Commodity VR Interaction Device for Gestural Object Manipulation in a Three Dimensional Work Environment*, ProQuest Dissertations Publishing, 2014.

- [27] "User Interface." *Cryengine*. N.p., n.d. Web. 25 Aug. 2016.  
<<http://docs.cryengine.com/display/SDKDOC2/User+Interface>>
- [28] "User Interfaces for VR." *Unity*. N.p., n.d. Web. 25 Aug. 2016.  
<<https://unity3d.com/learn/tutorials/topics/virtual-reality/user-interfaces-vr?playlist=22946>>
- [29] "Virtual Reality Company, ImmersaCAD, Unveils New Website, App." *Journal of Engineering*, 2016., pp. 182.
- [30] Xia, PingJun, António M. Lopes, and Maria T. Restivo. "A Review of Virtual Reality and Haptics for Product Assembly (Part 1): Rigid Parts." *Assembly Automation*, vol. 33, no. 1, 2013., pp. 68-77doi:10.1108/01445151311294784.
- [31] Xia, Pingjun, António Mendes Lopes, and Maria T. Restivo. "A Review of Virtual Reality and Haptics for Product Assembly: From Rigid Parts to Soft Cables." *Assembly Automation*, vol. 33, no. 2, 2013., pp. 157-164doi:10.1108/01445151311306672.
- [32] Zadeh, Wang, and Kubica. "The effect of sub-threshold forces on human performance in multi-modal computer-aided design." *Computer-Aided Design*, vol. 42, 2010, pp. 471-477.
- [33] Zhao, Long. *Development of a VR – Based CMM System for Industry Training and CMM Path Planning*, ProQuest Dissertations Publishing, 2013.