Gesture-based user interactions for product design review

by

Yu Xiao

A Thesis submitted to the Faculty of Graduate Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Mechanical Engineering

Faculty of Graduate Studies

University of Manitoba

Winnipeg, Manitoba

August 2017

Abstract

Human-computer interactions (HCI) are essential in computer-aided design (CAD) systems. Replacing the traditional computer mouse and keyboard by gestures for the design input has aroused wide interests of researchers to improve the naturalness and intuitiveness of HCI. Gesture-based design review systems are developed in this thesis for the CAD model review. Body gestures and hand gestures are captured using Microsoft Kinect and Leap Motion Controller, respectively. A template-based method is applied for the gesture recognition with the average gesture recognition rate of over 80%. Three of the frequently-used CAD commands including translation, rotation and scaling are proposed using gestures for the design review process. Applications of the design review systems show that the proposed methods are able to effectively trigger required design review operations via gestures. Results of the user tests show that intuitiveness and naturalness of HCI are improved via gestures compared to traditional methods of the design input. Users can have a better understanding of the product design using body gestures for the assembly review, and hand gestures for the detail review.

Acknowledgments

I would like to take this opportunity to thank my advisor, Dr. Qingjin Peng. Without his constant support and great patience, I would not be able to complete this thesis. His encouragement is very important for me to complete my degree. I wish to acknowledge that this research has been supported by the Discovery Grants from the Natural Sciences and Engineering Research Council (NSERC) of Canada and GETS program from the University of Manitoba. I would also like to give my thanks to my colleagues in the Virtual Manufacturing Lab and friends who helped me for my research and thesis. Finally, I would like to thank my parents. They are the most important people in my life. Without their support and understanding, it is impossible for me to obtain this degree.

Contents

List of Tables

List of Figures

List of Appendices

Chapter 1

Introduction

1.1 Research background

Engineering tools play important roles in product design and validation. Tools such as computer-aided design (CAD) and computer-aided engineering (CAE) are integral in product development. Though these tools are effective in creating, modifying, and evaluating designs, it is not natural and intuitive to use these tools with the computer mouse and keyboard for the human computer interaction (HCI). In a design process, an intuitive and natural interactive way is desirable for users, such as using gestures in a design process. Design review is an important procedure of the product development. Its goal is to review product design to evaluate outcomes of a design and to identify problems before manufacturing, such as design errors and manufacturing difficulties. One important aspect in the design review is the understanding of design solutions. Typically, a number of formal and informal reviews are conducted during a design project. An efficient interface is needed to understand complex 3D geometries, component details in an assembly, etc. Interaction devices like the Microsoft Kinect (Zhang, 2012) and Leap Motion Controller (Leap Motion Controller, 2015) could help improving the naturalness and intuitiveness of CAD and CAE tools by introducing gesture-based interfaces for HCI.

Gestures are commonly used as nonverbal communications between humans (Pavlovic, Sharma and Huang, 1997). As an intuitive and interactive way of communication, a number of human-computer interaction research activities have been conducted since the 1980s (Billinghurst, 1998). HCI interfaces are generally in the class of natural user interfaces (NUI) in which a variety of interactive possibilities are offered. The naturalness and intuitiveness are two major goals to achieve effective interactions between computers and human. We adopt the definition of "naturalness" as being easy to learn and remember and "intuitiveness" as the natural understanding of interactions (Grandhi, Joue and Mittelberg, 2011).

Many HCI interfaces have also been developed in the design field. There are studies using hand data gloves (Kumar, Verma and Prasad, 2012), and wrist-worn gloveless sensors to enhance human interactions in design (Kim *et al.*, 2012). However, as the special wearable hardware, there is an obstacle for them to be widely accepted. Availability of the low cost vision-based body gesture tracking devices such as Microsoft Kinect have spurred widespread interests in using gestures as the input of computers (Ren *et al.*, 2013). Microsoft Kinect is selected as the body gesture sensor in this research based on following three reasons. Firstly, Kinect is a cost-effective device. The price of professional motion tracking systems such as IMU, OptiTrack is 10 or 20 times more than Kinect. Kinect does not require users to wear special devices for motion tracking. Secondly, Kinect can capture human skeleton joint data with the acceptable accuracy compared to other motion tracking systems (Yeung *et al.*, 2014). Thirdly, unlike professional tracking systems with complicated parts, Kinect is easy to use for interacting with computers using body gestures.

Apart from using body gestures as the input of computers, designers can also interact with product models using hand gestures and motion tracking technologies. Different devices are available for the hand motion tracking including contact and non-contact sensors. There were studies using contact devices such as hand data gloves and wrist-worn gloveless sensors to detect the motion of hands (Kumar et al., 2012; Kim et al., 2012). However, non-contact devices have less hindrance to the hand motion compared to contact devices. Microsoft Kinect can track the human skeleton with a software development kit (SDK). There are studies for the hand gesture recognition using depth data from the Kinect sensor mainly focus on static gestures (Fiorentino et al., 2012; Vinayak et al., 2013; Le et al., 2014). Due to limitations in the accuracy and resolution of the device, Kinect is not a suitable device for detecting the hand motion. Recently, the Leap Motion Controller (LMC) has been developed to track the hand motion. The LMC is selected as the hand gesture recognition sensor in this research. The LMC as a hand motion tracking device has the following advantages. It is a vision-based tracking device with the low cost. Hand motions can be tracked in an interaction zone that is an inverse pyramid area up to 600 mm. Captured data can reach the accuracy of 200 μm (Weichert *et al.*, 2013). The LMC is explicitly targeted for hand tracking, and the orientation of hands and the position of fingers are computed automatically.

The existing CAD systems mainly use the computer mouse and keyboard as input devices. A natural interface with an intuitive way can improve the user experience and involvement and increase understanding of CAD models. Virtual Reality (VR) is a technique that utilizes the computer graphics and special input/output devices to generate immersive and interactive environments for users. With advanced 3D visualization capabilities, VR shows superior performances with a new perspective for users to interact with CAD models. It can enhance users' immersive feeling and depth perception of 3D objects. Therefore, using devices like the Microsoft Kinect and LMC via gestures as the design input in VR environments offers users effective interactions with product models in a more natural and intuitive way than that using the computer mouse and keyboard.

Incorporating gesture manipulations into CAD systems does introduce changes to the user experience. However, two of the following aspects are essential to be considered to design a user-friend, natural and intuitive design review system: the evaluation of devices and comparison of gestures. The evaluation with devices is required to promote the development of gesture-based HCI. The comparison of hand and body gestures for different review purposes will be helpful to decide the user preference for creating a user-friendly interface.

Generally speaking, engineering CAD and CAE tools play a crucial role in designing and evaluating products. However, the ease of use, naturalness and intuitiveness are limited due to the traditional mouse and keyboard as input tools for the interaction. The work presented in this thesis evaluates two of the gesture-based interaction devices and compares features of different design review systems.

1.2 Research objectives

The objective of this research is to establish a gesture-based design review system in an immersive VR environment to allow users to review the product assembly and details effectively and to increase the understanding of product design in a natural and intuitive way. Based on literature review of gesture recognition methods and HCI applications, required functions for the gesture-based design review system are decided. An immersive VR environment is integrated with the gesture-based design review system to improve the user experience. Static and dynamic gestures are designed to replace the computer mouse and keyboard for the design review command input. Three of the commonly-used operations in most CAD systems including translation, rotation and scaling are selected for the design review system. A template-based matching method is applied to achieve gesture recognition to guarantee that designed gestures can be mapped with extracted data from the sensors. The performance of HCI devices Microsoft Kinect and LMC are evaluated by gesture recognition experiments. The user experience is investigated by the user test with the statistical analysis in reviewing product design in the assembly and detail.

In this thesis work, there are three main contributions: (1) A body gesture-based CAD design review system in a VR environment using the Microsoft Kinect is proposed. (2) A hand gesture-based design review system is developed using the Vizard VR software (Vizard, 2015) and LMC. (3) An integrated body and hand gesture-based design review system is established with the integration of Microsoft Kinect and LMC to achieve reviewing the product assembly and detail. Comparisons of these three systems and the mouse and keyboard as inputs for the design review are made through user tests and statistical methods.

1.3 Thesis contents and structure

As shown in Figure 1-1, this thesis is organized as follows. This Chapter introduces the research background, research objectives and the thesis outline. Chapter 2 introduces HCI devices, virtual reality (VR), gesture recognition methods and related research applying the gesture recognition in HCI. The statistical analysis of user experience questionnaire is also reviewed. The body gesture-based design review system is introduced in Chapter 3, which includes the structure of the proposed system, designed gestures and manipulation, system application and evaluation. Chapter 4 involves a hand gesture-based interface for the detail design review. The constitution of the proposed system is described. Hand gestures are designed for the design review commands and hand gesture templates are applied for the gesture mapping. A shaft model with eight detail components and a wheel model with seven detail components are used for the application of the proposed system and system evaluation. The integration of body and hand gestures for the design review is presented in Chapter 5. The Microsoft Kinect is used for the body gestures recognition and assembly review, and the LMC is applied for the hand gestures recognition and detail review. An application is described to verify the proposed system. The advantages and disadvantages of proposed three design review systems in Chapters 3, 4 and 5 are compared. Chapter 6 concludes the thesis and identifies contributions of the research. Future work is also discussed.

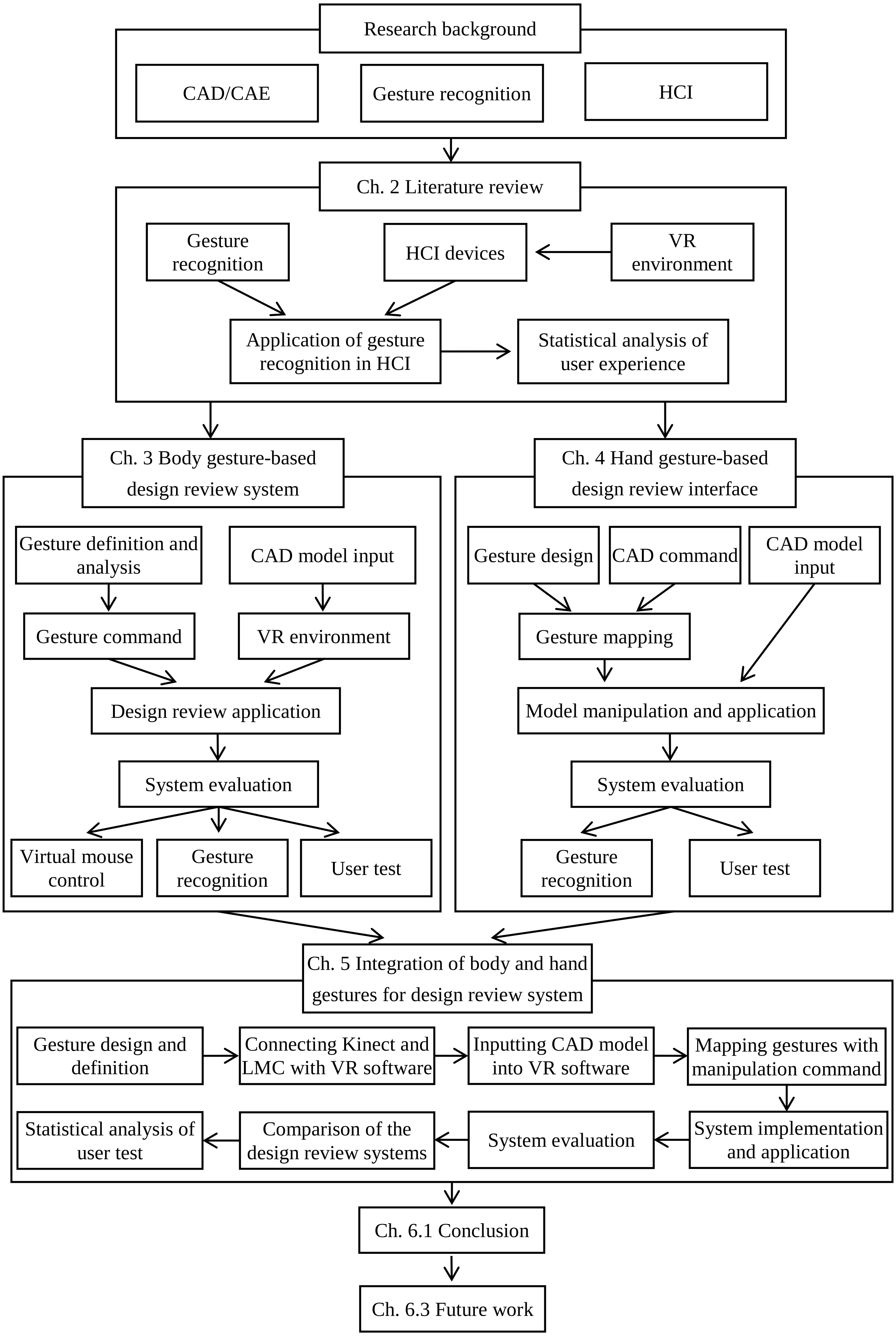


Figure 1-1 Thesis outline

Chapter 2

Literature review

2.1. Human computer interaction (HCI) devices

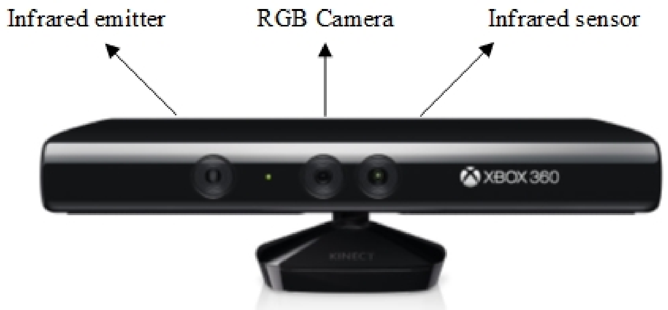
Though advanced engineering tools and technologies such as CAD and CAE have emerged for conducting, improving, modifying and evaluating the product design, there is still a lack of investigation making the human computer interaction more natural and intuitive. The traditional devices such as the computer mouse and keyboard are not so natural and intuitive for HCI. Gesture-based HCI interfaces can be adopted to enhance the naturalness and intuitiveness of HCI. Contact-based devices and noncontact-based devices are the two kinds of devices for the gesture recognition and motion capture.

The contact-based devices are best represented by the data glove such as CyberGlove (Fels, Pritchard and Lenters, 2009; Kessler, Hodges and Walker, 1995; Mohandes, 2013), Powergloves (Kadous, 2002), and CyberGloves II (Kováč, Ďurovský and Varga, 2014). Though these glove sensors can accurately measure the spatial positions and hand joint angles, wearing gloves make gesturing become cumbersome. To make the gloves less cumbersome, colored markers are applied in acquiring spatial positions of certain fingertips (Mistry and Maes, 2009; Wang and Popović, 2009). For the motion capture and body gesture recognition, contacted-based professional tracking systems are applied such as IGS-190, IMU and

OptiTrack (Ishii *et al.*, 2011; Kim *et al.*, 2009; Chang *et al.*, 2012). However, the contact-based devices require users to wear sensors which hinder motions of the body and hand.

Comparing with contact-based devices, noncontact-based devices provide the more freedom of movements. These devices are introduced using the RGB camera, depth camera and infrared camera (IR camera). They usually include one or more cameras. RGB cameras are used for the hand gesture recognition based on the difference of skin colors (Shin, Tsap and Goldgof, 2004; Palacios *et al.*, 2013). One limitation of RGB cameras is the sensitivity to light conditions. Changes of the light conditions can affect the image and further segmentation of the hand under the image background. Devices installing the depth camera or IR camera such as Microsoft Kinect and Leap Motion Controller (LMC) can reduce the effect of light conditions for the gesture recognition and motion capture.

The Kinect sensor shown in Figure 2-1 contains a depth camera, a color camera and a four-microphone array that provides full-body 3D skeleton tracking. The color camera of the first generation of the Kinect sensor has a resolution of 640×480 pixel at 30 Hz. The depth camera has a resolution of 640×480 pixel at 30 Hz. Many studies use Kinect for the full body motion capture and body gesture recognition (Dave *et al.*, 2013; Fiorentino *et al.*, 2012; Cassola *et al.*, 2014; Hsieh *et al.*, 2014). It can track 20 skeleton joints.



SPECS
- Video Camera: 640 x 480 Pixel/30fps
- Depth Camera: 640 x 480 Pixel/30fps
- Range: 0.8-4.0m
- Horizontal Field of View: 57degree
- Vertical Field of View: 43degree

Figure 2-1 Microsoft Kinect (*Kinect Robotic*, 2012)

Instead of full-body tracking, the LMC shown in Figure 2-2 specializes in hand motion tracking with a higher resolution. The LMC has two Infra-Red cameras (IR cameras) and three IR emitters, which can detect hands both in bright and dark environments (Leap Motion Controller, 2015). Study on the accuracy of sensors suggests that it can be an effective tool for detecting hand gestures (Guna et al., 2014). Many researchers applied LMC as a tool for the gesture recognition and object manipulation (Potter, Araullo and Carter, 2013; Kerefeyn and Maleshkov, 2015; Avola et al., 2014).

Figure 2-2 Leap motion controller (*Leap Motion*, 2015)

There are different kinds of sensors launching these years with their own features for the motion capture and gesture recognition. The work presented in this thesis develops and evaluates design review systems with the Kinect sensor and LMC for body and hand gestures.

2.2. Virtual reality (VR)

VR allows users to see and review their design in an immersive way. Combining VR with CAD and CAE provides users a better understanding of product design. Berta compared features of the CAD and VR systems with the potential benefits of the integration of VR and CAD/CAE systems (Berta, 1999). The benefits include creating simple and natural interfaces for users and the immersive sense of feeling. Tseng et al. summarized the proper research applications using VR for the immersive 3D visualization (Tseng *et al.*, 2017). Immersion is defined as a complete involvement in the virtual environment (Schuemie *et al.*, 2001). Mujber et al. presented an overview on VR applications in manufacturing processes (Mujber, Szecsi and Hashmi, 2004). They categorized applications of VR in the manufacturing process into three areas as shown in Table 2-1.

Table 2-1 VR applications (Mujber, Szecsi and Hashmi, 2004)

Area	Design	Operation management	Manufacturing
Application	Design	Planning	Machining
	Prototyping	Simulation	Assembly/Disassembly
		Training	Inspection

Barfield et al. investigated the effects of stereopsis and head tracking on visualizing the structure of objects in a desktop virtual environment (Barfield, Hendrix and Bystrom, 1997). They find that head tracking plays an important role in virtual object visualization. Wang et al. presented a manual assembly design system with enhanced user experience using AR technologies (Wang, Ong and Nee, 2013). An assembly data structure is designed for assembly information management and a hybrid approach has been formulated and implemented to allow users to simulate a manual assembly process. A 3D bare-hand interaction is integrated with the assembly design for users to manipulate virtual components in a natural and effective way. Freeman and Coburn presented a VR interface for the CAD design visualization in an assembly hierarchy (Freeman and Coburn, 2016). CAD design can be reviewed in the Unity software. Design parameters can be modified using interaction tools in the VR environment. EI-Mounayri et al. proposed an educational tool for the operation of a computerized numerical control (CNC) milling machine utilizing a virtual environment (EI-Mounayri, Rogers and Fernandez, 2016). They find that the environment is realistic and easy to navigate and the immersive technology is beneficial for the educational purpose. Fechter et al. introduced a VR-CAD assembly system for the product developer to achieve assembly planning (Fechter *et al.*, 2014). They combined the highly intuitive interaction tools and VR techniques for visualization. A case study of a gear box assembly process is implemented. Pontonnier et al. simulated the assembly process of a digital mock-up (DMU) in both real and virtual environments (Pontonnier *et al.*, 2014). An evaluation of real environment (RE), virtual environment (VE), and virtual force feedback (VFF) environments is provided. Results indicate that there is a force sensory gap between RE and VFF, they address this problem using haptic devices for simulation. Fillatreau et al. developed a unique framework for the checklist-based project for all steps of the Product Lifecycle Management (PLM) in an immersive VR environment (Fillatreau *et al.*, 2013). At different stages of the PLM, companies develop numerous checklist-based procedures involving the prototype inspection and testing. The framework combines an immersive navigation in the checklist, virtual experiments when needed and multimedia update of the checklist. Song et al. applied the VR technology for an open-architecture product design evaluation process (Song *et al.*, 2017). The interactive system provides users a close-real experience in the evaluation process. Pilia et al. presented the application of VR tools for assembly of a fusion machine (Fillatreau *et al.*, 2013). The simulation process makes it possible to be aware of the real size of a component and future difficulties in a real assembling process. The simulation is evaluated and compared to the physical mockup for the enhancement of VR tools. The aim of the application is to build a design tool that helps the designers from early stage of the designing process by taking into consideration of the assembly and maintenance aspects to reduce the project costs and the time of developing period. The research indicates that the VR technique is a useful tool in the designing and manufacturing simulation. Users can manipulate and interact with product models in an immersive environment for a better understanding of the product design.

2.3. Methods of the gesture recognition

It is a complex task to recognize human gestures and map them into specific commands. There are different methods for the gesture recognition such as the template-based gesture recognition (Wobbrock et al., 2007; Nguyen-Dinh et al., 2012) and the machine learning-based gesture recognition including Support Vector Machine (SVM) (Marin et al., 2014), k-Nearest Neighbours (KNN) (Nagarajan and Subashini, 2015; Artyukhin and Mestetskiy, 2015) and Hidden Markov Models (HMM) (Mccartney et al., 2015). The classification of gesture recognition methods are shown in Table 2-2.

The template-based gesture recognition is also known as the pattern-based gesture recognition. The gesture recognition engine matches users' movements with predefined gesture templates. In this approach, gestures are first recorded and stored as the gesture templates. During the matching process, the performed gestures are taken as input and will be validated against the stored gesture templates. The template-based gesture recognition system as shown in Figure 2-3 involves three phases: creating gesture templates, tracking gesture and extracting features, template matching.

The HMM is a common method for the dynamic gesture recognition represented by a set of finite states with their transitional relationships characterized by the state transitional probabilities. The SVM and KNN are statistical classifiers. The prior one can deal with both linear and non-linear classifications by mapping inputs into high-dimensional feature spaces, and the latter is a simple algorithm that stores samples and classifies new data based on a distance similarity measure. However, a large amount of training samples is needed for these classifiers. Considering the small set of design gestures and the gesture recognition accuracy, the template-based gesture recognition is applied in this research (Pradipa and Kavitha, 2014).

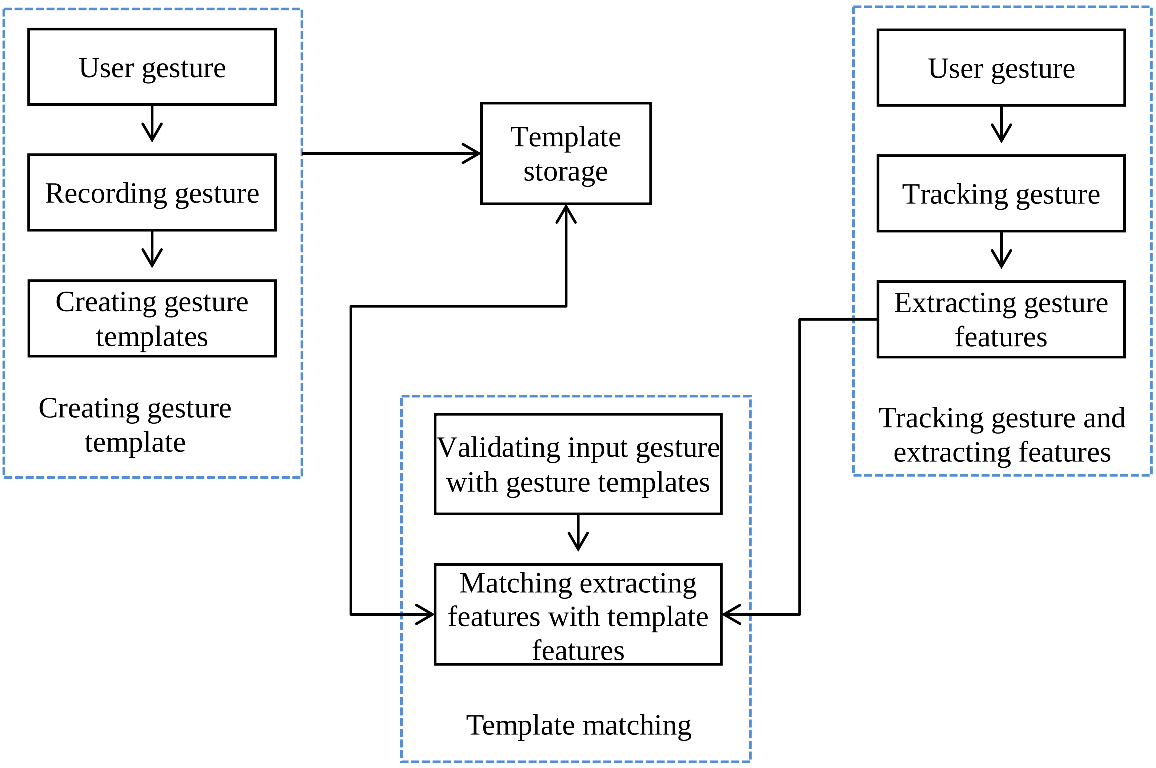


Figure 2-3 Template-based matching process

Table 2-2 Classification of gesture recognition methods

Method	Gesture type	Reference
Template-based	Finger gesture	(Wobbrock, Wilson and Li, 2007)
Template-based	Finger gesture	(Nguyen-Dinh *et al.*, 2012)
Template-based	Body gesture	(Mahbub *et al.*, 2013)
Template-based	Body gesture	(Camgoz, Kindiroglu and Akarun, 2015)
Template-based	Hand gesture	(Ren *et al.*, 2013)
Template-based	Hand gesture	(Carrera *et al.*, 2014)
Template-based	Hand gesture	(Wang and Wang, 2016)
SVM	Hand gesture	(Marin, Dominio and Zanuttigh, 2014)
SVM	Body gesture	(Miranda *et al.*, 2012)
kNN	Hand gesture	(Nagarajan and Subashini, 2015)
kNN	Hand gesture	(Artyukhin and Mestetskiy, 2015)
HMM	Finger gesture	(Mccartney, Yuan and Bischof, 2015)
HMM	Hand gesture	(Keskin, Erkan and Akarun, 2003)
HMM	Body gesture	(Ding and Chang, 2016)

2.4. Applications of the gesture recognition in HCI

With the development of gesture recognition technologies, researchers have been interested in applying the gesture recognition in HCI since the past decades. Kumar et al. applied the data glove for painting and writing characters in a real-time environment (Kumar, Verma and Prasad, 2012). Stoerring et al. utilized a head mounted device (HMD) and head mounted camera (HMC) for the gesture recognition in an augmented reality environment (Stoerring *et al.*, 2004). Wingrave et al. reviewed the application of using the WII remote controller to develop a 3D dance interface (Wingrave et al., 2010). Kim et al. utilized the wrist-worn gloveless sensor to build a virtual hand model applied in the 3D spatial interaction with mobile devices (Kim et al., 2012). After the release of difference gesture recognition devices, the performances of the devices in HCI have been analyzed in different research.

Buchmann et al. introduced a gesture-based system for the direct manipulation of virtual objects (Buchmann *et al.*, 2004). They attached fiducial markers on each finger to track the fingertips and to derive gestures. Their systems allow users to pick up virtual objects in a 3D space. Francese et al. presented a research using the Wii mote and Microsoft Kinect to help people explore 3D geographical maps using body gestures (Francese, Passero and Tortora, 2012). Participants navigated the maps using Wii mote and Kinect separately, and their reactions were recorded. Results of the study showed that motion control devices increased the users' sense of immersion and presence during the interaction process. The gestural interfaces quickly bring the users from novice to expert navigation operations. Shiratuddin and Wong developed a gesture-based interaction paradigm with a non-contact gesture recognition system to detect real-time hands and fingers movements and their spatial positions in the 3D space (Shiratuddin and Wong, 2011). These gestures are then interpreted and executed for specific commands in a virtual world for architectural design. Fiorentino et al. created a system that users can explore a CAD model through an augmented reality environment and simulate the assembly process (Fiorentino *et al.*, 2012). The authors performed a user study to validate the interface for users to become proficient and comfortable exploring CAD models. Tumkor and Esche used two Kinect sensors for users to explore and disassemble CAD models using hand gestures (Tumkor and Esche, 2013). To evaluate the system, the researchers conducted a user study measuring the completion time of assembling a design model. From the study they found that using gestures to perform some certain operations was faster than using the computer mouse and keyboard. Gallo et al. explored the free hand navigation to control medical imaging data using Kinect (Gallo, Placitelli and Ciampi, 2011). In their work, they used gestures for manipulating medical models. Dave et al. developed a gesture interface for 3D CAD modeling using Kinect (Dave, Chowriappa and Kesavadas, 2013). Scale, rotate, translate commands are designed for manipulating 3D models. Nanjundaswamy et al. created a system that incorporates gestures, brain-computer interface and speech to make an interactive CAD system (Nanjundaswamy *et al.*, 2013). They combined these three elements to make the traditional CAD systems more intuitive beyond using standard mouse and keyboard. Song et al. developed a system for interacting with CAD models using gaze and finger control (Song *et al.*, 2014). They identified three primary CAD tasks in order to build and test their system for translation, rotation, and zooming. Xu et al. proposed a non-touch volume interaction prototype using the LMC (Xu *et al.*, 2015). They proposed a 3D volume interactive interface for the medical image visualized in different layers. Lee et al. applied gestures to control content displayed on a television screen (Lee *et al.*, 2013). They used Wizard of Oz studies to develop gesture sets and allow user evaluation of gestures. They thought that it was better to focus more on how users would complete a task and less about how well a system would capture a certain gesture. Hsieh et al. presented a VR system to prevent elderly from falling down combining Unity3D with Kinect, and different body gestures were analyzed (Hsieh *et al.*, 2014). Skeleton joint data from Kinect are used to evaluate participant's body condition in the training process. Sabir et al. developed an intuitive system that allowed biologists to explore molecular structures (Sabir and Tabor, 2013). The researchers conducted a user study. Babu et al. described a 3D sketch-based system for users to freely sketch 3D shapes using gestures

(Babu *et al.*, 2014). The system can automatically classify sketches drawn in a 3D environment with the predefined sketch set for the sign language recognition.

Gesture recognition has been applied in different HCI applications. Low cost devices like Microsoft Kinect and LMC with proper software development kits (SDK) have made them accessible for research. From the work mentioned above, we can see that the potential of the gesture recognition is noticeable. However, combining gestures with proper applications in the CAD/CAE area is still under development.

2.5. Statistical analysis of user experience questionnaire (UEQ)

Comparing the new HCI interfaces with traditional mouse and keyboard interface is a common method to evaluate the effectiveness and performance of the new interface (Tang, Lee and Gero, 2011; Francese, Passero and Tortora, 2012). A widespread procedure is based on comparative methods for analyzing the user performance of doing certain tasks in different interfaces. A comparing procedure was labeled by Buisine as the paradigm evaluation for comparing independent groups realizing the same activity (Buisine *et al.*, 2012). Different factors are measured in the paradigm evaluation such as the attractiveness, efficiency, novelty, etc. A common investigation concerns users' opinions toward the quality of new features comparing to the old. Users' opinions are usually measured through questionnaires based on Likert and Likert-type items in rating scales.

The results of the questionnaires are analyzed to find the statistical differences between the compared aspects. In this research, the questionnaire results of the user test are analyzed following the guideline proposed by Guerra et al (Guerra, Gidel and Vezzetti, 2016). The guideline aims at providing a pragmatic common procedure to analyze Likert and Likert-type questionnaire result in a scientifically-rigor and comparable way. The guideline is shown in Table 2-3. Five types of problems can be analyzed following the guideline: estimate sample size, estimate effect size, estimate statistical power, estimate statistical significance for independent groups and estimate statistical significance for related groups. In this study, the analysis is focusing on estimating the statistical significance for independent groups because there are no correlations among the different design review systems in operating process. T-test, z-test and ANOVA are included in parametric methods, and Mann-Whitney is included in nonparametric methods.

Table 2-3 Questionnaire analyzation guideline (Guerra, Gidel and Vezzetti, 2016)

Type of problem	**Parametric methods**	**Nonparametric methods**
Estimate sample size	Through Z-score table	Parametric sample size*1.15
Estimate effect size	Risk difference, risk ratio, odds ratio, Cohen's d, Glass's delta, Hedges' g, the probability of superiority	Cliff's delta
Estimate statistical power	Equal to $1-\beta$. Use Cohen's power table.	Monte Carlo simulations
Estimate statistical significance for independent groups	t-test, z-test, ANOVA	Mann-Whitney
Estimate statistical significance for related groups	Paired t-test, z-test	Wilcoxon rank sum test, sign test

This chapter described the development of HCI devices, applications of VR, gesture recognition methods, applications of the gesture recognition and statistical analysis of UEQ. Microsoft Kinect and LMC are selected as sensors for the body and hand gesture recognition because of their low cost, contact-free and accuracy. The VR technology is proved to be a natural interface for users with an immersive sense of feeling. Vizard VR software is chosen as a platform for the product design review. The template-based and machine learning-based gesture recognition methods are summarized. Though the machine learning-based method shows the great performance in the gesture recognition accuracy, the template-based method is chosen for the gesture recognition because its reliable gesture recognition accuracy for a small set of design gestures without capturing training samples. The existing research has improved the gesture recognition accuracy using gesture recognition methods and devices, including comparing the device performance, testing gesture recognition methods with a large amount of design gestures. However, most researchers focused on the gesture recognition accuracy, and the research solutions regarding gesture recognition applications are limited in recognizing sign language and manipulating CAD models in assembly. This research evaluates two of the gesture-based interaction devices. The traditional computer mouse and keyboard as input devices are compared with gesture inputs and features of different design systems for the assembly review, detail review and comprehensive review in the product assembly are analyzed for the better understanding of product design. User test results are studied using the statistical method.

Chapter 3

Body gesture-based user interaction in design review

3.1. Structure of the proposed system

The proposed system in this research consists of three parts including the CAD model input, VR system and Microsoft Kinect device. Their relationships are shown in Figure 3-1. The CAD model input, based on the neutral format of CAD software, renders models for compatible data formats used in the VR system. The detail of CAD model input method is shown in Figure 3-2. The OSGB or Open Scene Graph Binary data format is used in the propose system. The VR system, based on Python programming, is a platform to visualize and navigate product models in the VR environment. Product models can be translated in x, y, z axes, scaled up and down, rotated in x, y, z axes using the body gesture as the input. Kinect using the camera and depth sensor is used as a tool to capture human skeleton joints' data through a third party library-OpenNI (Zhang, 2012). It allows the middleware FAAST developed based on OpenNI and Microsoft SDK libraries to pass Kinect data of 24 skeleton joints to the VR system for a real time motion tracking (Suma *et al.*, 2013). Figure 3-3 shows a screenshot of the body skeleton tracking using Kinect.

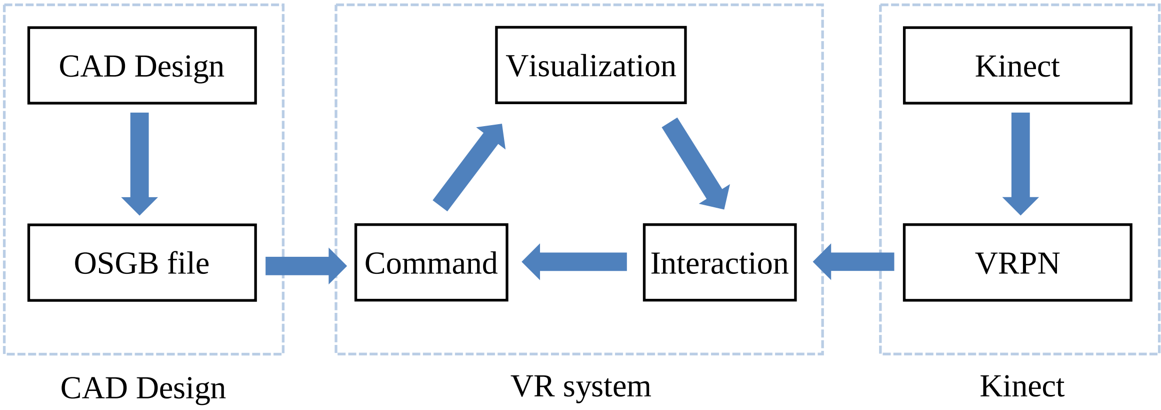


Figure 3-1 Proposed system structure

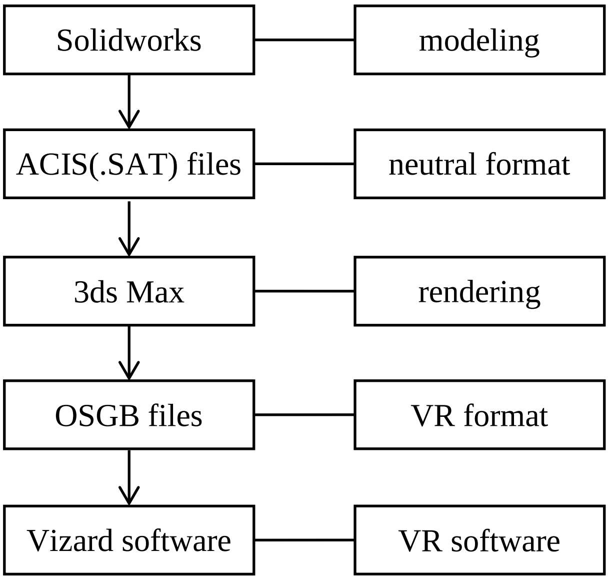


Figure 3-2 CAD model input

Figure 3-3 Kinect skeletal tracking

3.2. Gestures and manipulations

3.2.1. Gesture definition

There are three types of commonly-used manipulation tasks in most CAD systems including translation, rotation and scaling (Song *et al.*, 2014). These tasks are usually operated using the computer keyboard and mouse. Using gestures to interact with computer gives users a reality feeling with new experience. An intuitive and natural way to carry out these tasks is introduced in this research using body gestures.

The scaling, rotation and translation are three frequently used commands in a design review process. Apart from translation, rotation and scaling commands, navigation, exploding and assembly commands are added to the system. Six types of CAD tasks are linked to gestures in the proposed system as shown in Table 3-1.

Considering the human cognitive load, some of the proposed gestures are used for multiple commands depending on the selection of the operation menu (Thakur and Rai, 2015). The body gesture design refers to Shiratuddin's research of non-contact interactive systems for the architectural design (Shiratuddin and Wong, 2011). For static gestures, six gestures are defined to represent movements in the navigation and translation commands. Users can view the model from different directions and translate their positions. A coordinate system of the interface is shown in Figure 3-4. For dynamic gestures, six basic gestures are defined with combinations of two in a certain sequence. If a user performs gestures in the defined sequence, it will trigger the related command. The gesture for scaling up is the same as disassembly and the gesture for scaling down is the same as assembly.

Table 3-1 Gesture definition

Static Gesture		
	Gesture: Move forward Category: Navigation When the right hand is raised forward and the arm is around 90 degree to the chest, the main view will be navigated forward.	Gesture: Move forward (Z axis positive) Category: Translation When the right hand is raised forward and the arm is around 90 degree to the chest, the selected part will be translated forward.
	Gesture: Move backward Category: Navigation When left and right hands move backward together, the main view will be navigated backward.	Gesture: Move backward (Z axis negative) Category: Translation When left and right hands move backward together, the selected part will be translated backward.
	Gesture: Move upward Category: Navigation When the left hand is raised up and the elbow to the upper arm is around 90 degree, the main view will be navigated upward.	Gesture: Move upward (Y axis positive) Category: Translation When the left hand is raised up and the elbow to the upper arm is around 90 degree, the selected part will be translated upward.
	Gesture: Move downward Category: Navigation When the left hand is raised down and the elbow to the upper arm is around 90 degree, the main view will be navigated downward.	Gesture: Move downward (Y axis negative) Category: Translation When the left hand is raised down and the elbow to the upper arm is around 90 degree, the selected part will be translated downward.
	Gesture: Turn right Category: Navigation When the right hand is raised up by the side and the elbow to the arm is around 60 degree to the right leg, the main view will be turned right.	Gesture: Move right (X axis positive) Category: Translation When the right hand is raised up by the side and the elbow to the arm is around 60 degree to the right leg, the selected part will be moved to the right side.

	Gesture: Turn left Category: Navigation When the left hand is raised up by the side and the elbow to the arm is around 60 degree to the right leg, the main view will be turned left.	Gesture: Move left (X axis negative) Category: Translation When the left hand is raised up by the side and the elbow to the arm is around 60 degree to the right leg, the selected part will be moved to the left side.
Dynamic Gesture		
 Gesture#1	 Gesture#2	Gesture: Scaling up Category: Scaling The selected model will be scaled up, when the user performs gesture#1 and gesture#2 in a sequence.
 Gesture#2	 Gesture#1	Gesture: Scaling down Category: Scaling The selected model will be scaled down, when the user performs gesture#2 first and follow by gesture#1.
 Gesture#3	 Gesture#4	Gesture: Rotate (X axis) Category: Rotation The selected model will be rotated in the X direction when user raises the left hand around the chest and follows by raising the right hand to X axis.
 Gesture#3	 Gesture#5	Gesture: Rotate (Y axis) Category: Rotation The selected model will be rotated in the Y direction, when user raises the left hand around the chest and follows by raising the right hand to Y axis.

 Gesture#3	 Gesture#6	Gesture: Rotate (Z axis) Category: Rotation The selected model will be rotated in the Z direction, when user raises the left hand around the chest first and follows by raising the right hand to Z axis.
 Gesture#1	 Gesture#2	Gesture: Explode Category: Explode The model will be disassembled automatically in order to provide an exploded view of different parts.
 Gesture#2	 Gesture#1	Gesture: Assembly Category: Assembly The selected model will return to the original place.

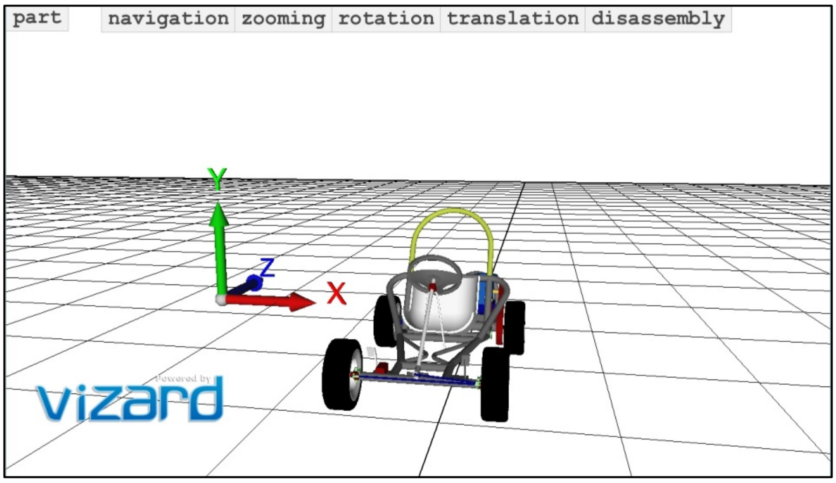


Figure 3-4 Design review interface's coordinate system

3.2.2. Gesture analysis

Recent developments of camera sensor technologies provide support to the gesture recognition based on skeleton data. Available platforms allow interactions in 3D virtual worlds using the motion capture technology, such as RINIONS, FAAST and NUILIB (Cassola *et al.*, 2014). These platforms for the data transmission between Kinect and the computer software simplify the process of using gestures in applications. FAAST (Figure 3-5) with the support of OpenNI and Microsoft SDK libraries is applied in this research as the middleware to transmit human skeleton joint data to the VR system.

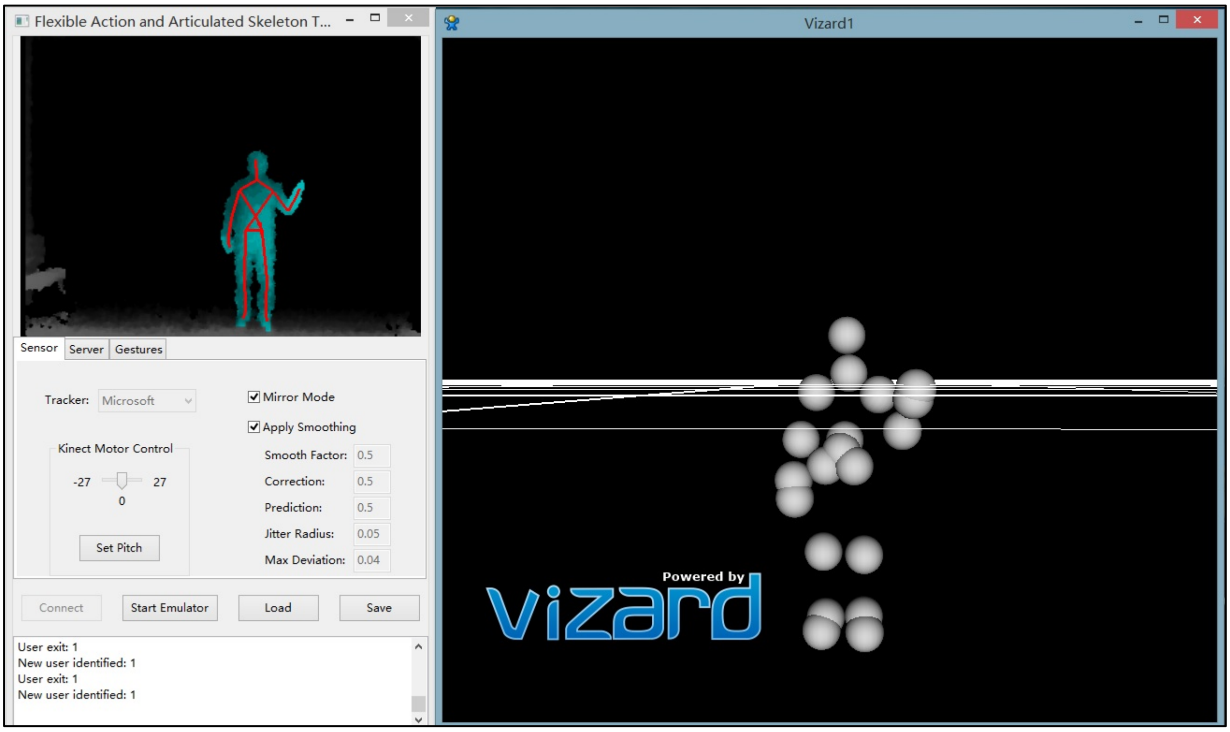


Figure 3-5 FAAST interface

The Kinect sensor captures human body skeleton data in the same scale of a real person and transmits the data into FAAST to integrate full-body skeleton tracking with VR applications. The VR system is connected to FAAST using the VRPN7 protocol (*VRPN*, 2016). FAAST streams user's skeleton data over a VRPN server and the server automatically starts when FAAST connects to a Kinect. A total of 24 skeleton joints' transformations are streamed. The default coordinate system in FAAST is compatible with the VR system. For matching gestures, users' gestures are compared to the relative positions between different skeletal joints (Sundar *et al.*, 2003).

A common skeleton matching method using skeleton joints is shown in Figure 3-6 (Sundar *et al.*, 2003). However, if the skeleton matching only uses these end nodes, different body gestures are likely to share very similar matching features, such as Figures 3-6a and 3-6b, Figures 3-6c and 3-6d. We can clearly notice that Figures 3-6a and 3-6b are totally different gestures, but the end node information is basically the same. Therefore, an improved method considering the topological structure of human skeletons is proposed in this research to reduce the position matching defect. An example of the topological structure of human skeletons is shown in Figure 3-7. Compared to the end nodes matching method in Figure 3-6, the topological structure of human skeletons can provide a more accurate body gesture matching process via setting up angular and distance relations between different skeleton joints in order to reduce the matching mistakes. The skeleton joints are represented by vectors as follows.

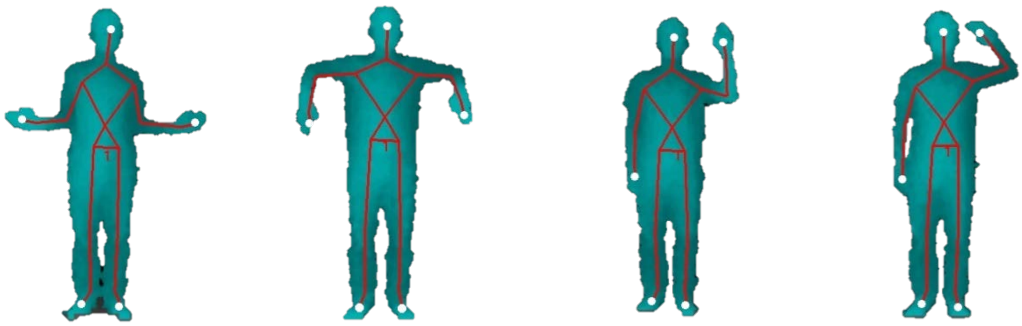


Fig. 3-5a Fig. 3-5b Fig. 3-5c Fig. 3-5d

Figure 3-6 Skeleton matching

Fig. 3-6a Fig. 3-6b

Figure 3-7 Topological structure of human skeletons

As the Kinect sensor tracks the human body skeleton from position number 0, where N is the number of skeleton joints, the corresponding vectors can be expressed as V = {v_0, v_1, v_2, …, v_N}. The spatial positions of individual joints can be expressed in Eqn. (3-1).

$$V_i = \{x_i, y_i, z_i\} \quad (i = 0, 1, 2, ..., N) \tag{3-1}$$

In order to set up the angular relationship between vectors, the new formed vector can be calculated using Eqn. (3-2).

$$\begin{aligned} \overrightarrow{v_{ij}} &= \overrightarrow{v_j} - \overrightarrow{v_i} = \{x_j - x_i, y_j - y_i, z_j - z_i\} \\ &= \{x_{ij}, y_{ij}, z_{ij}\} \quad \left(\overrightarrow{v_i}, \overrightarrow{v_j} \in V\right) \end{aligned} \tag{3-2}$$

The distance between two vectors is calculated using Eqn. (3-3).

$$D_{ij} = \overline{V_{ij}} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \tag{3-3}$$

The angle between two vectors is calculated using Eqn. (3-4).

$$\theta = \arccos \frac{\vec{v_i} \cdot \vec{v_j}}{\left|\vec{v_i}\right|\left|\vec{v_j}\right|} \tag{3-4}$$

Kinect sensor can detect 24 skeleton joints, it is possible to set up the angle and distance constraints among skeleton joints and their connections. As the coordinate system of the VR system is the same as the coordinate system of the Kinect sensor, the existing Kinect coordinate can be directly applied in the VR system as shown in Figure 3-8 (*Kinect Robotic,* 2012). The distribution of Kinect skeleton joints is shown in Figure 3-9.

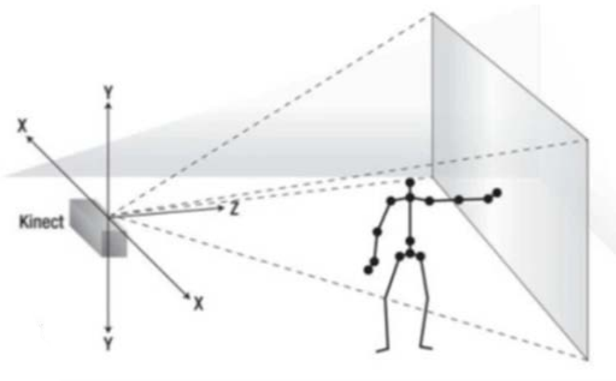


Figure 3-8 Coordinate system (*Kinect Robotic,* 2012)

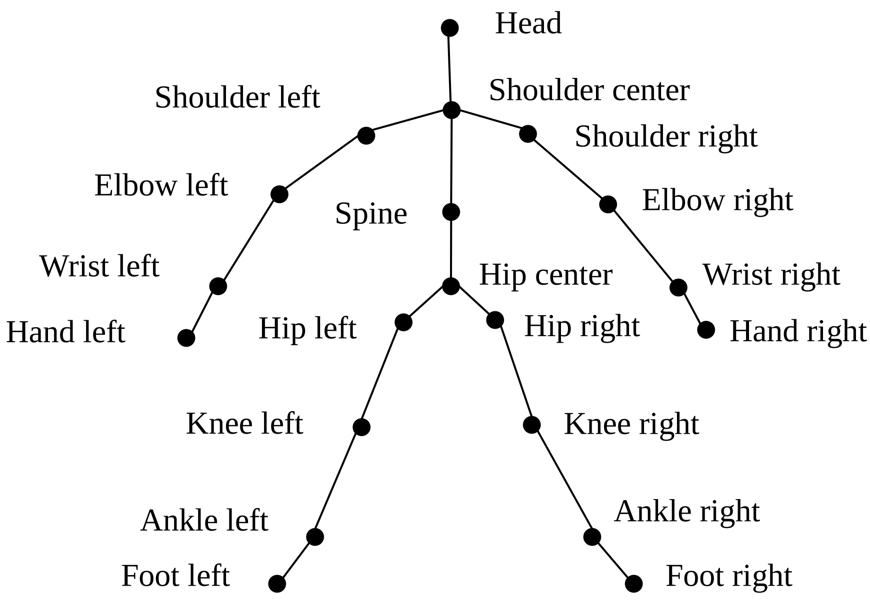


Figure 3-9 Kinect skeleton joints

The distance thresholds of different gestures are set up according to Eqn. (3-3) based on the vector distance calculation. However, the angle thresholds among different skeleton joints need to be adjusted. The vectors representing skeleton joints are shown in Figure 3-10.

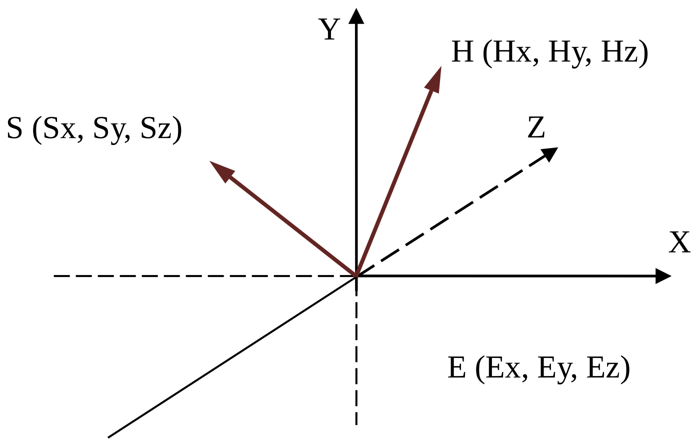


Figure 3-10 Skeleton joints in the 3D coordinate system

In Figure 3-10, S is defined as the shoulder, E is the elbow and H is the hand. Two new vectors have to be created in order to constrain the angle between “elbow to shoulder” and “elbow to hand”, the constraint angle θ can be expressed in Eqn. (3-5) as follows.

$$\theta = \arccos \frac{\overrightarrow{ES} \cdot \overrightarrow{EH}}{\left|\overrightarrow{ES}\right|\left|\overrightarrow{EH}\right|} \tag{3-5}$$

The turn left/move left gesture is selected as an example of gesture representation as shown in Figure 3-11. a1, a2, a3 in Figure 3-11 represent three attributes for turn left/move left gestures, a1 is the angle between the shoulder left to hand left and shoulder left to hip right, a2 is the angle between the elbow left to hand left and elbow left to shoulder left, a3 is the distance between the hand left and spine in Z axis. These three attributes form the left turning gesture template. When a1 is between 80 to 100 degrees, a2 is between 140 to 180 degrees and a3 is within 10 centimeters, the left turning gesture is recognized.

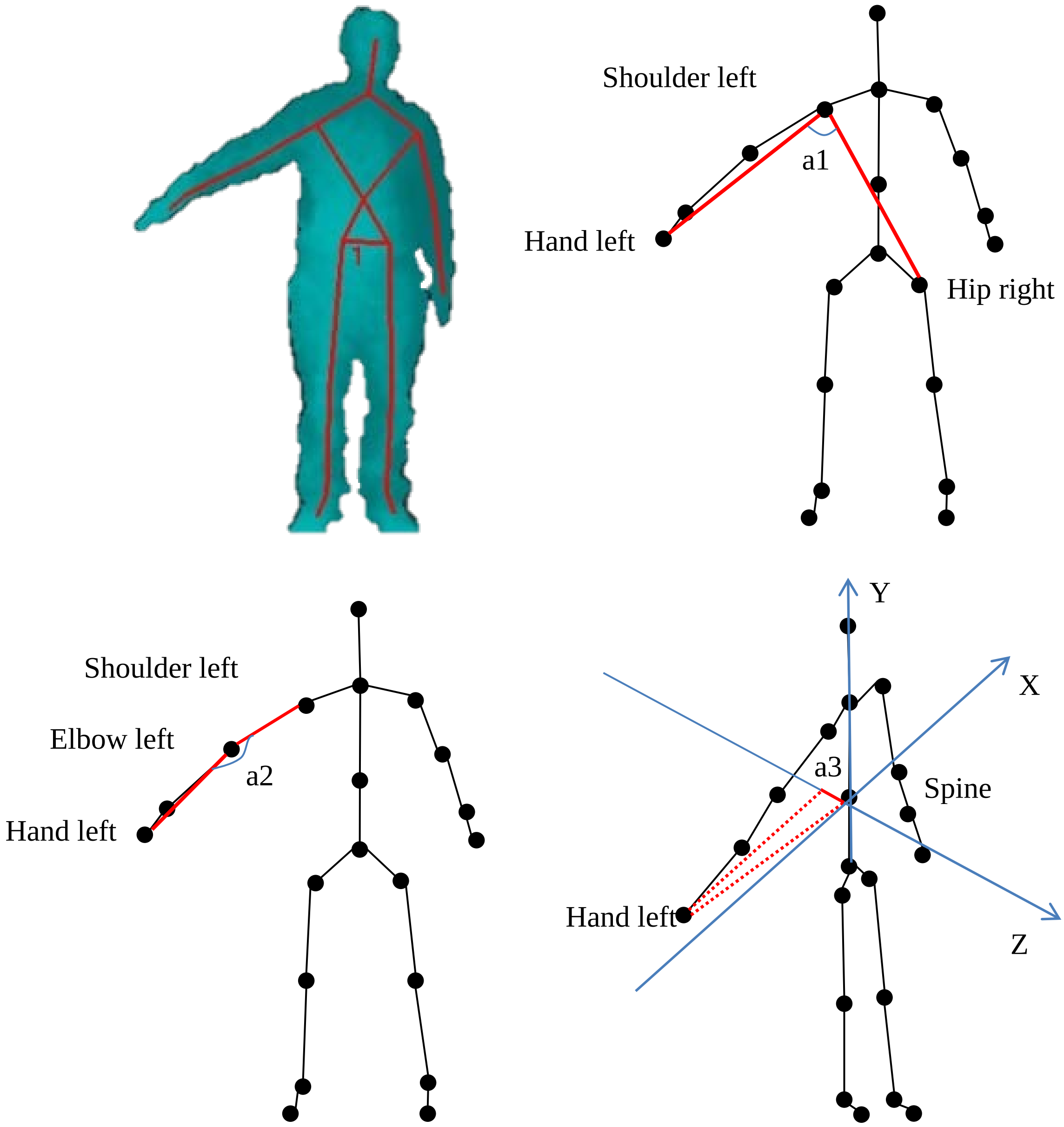


Figure 3-11 Example of gesture representation

3.3. Design review application

A user interactive interface is developed using the proposed system. It is assumed that a user stands in front of the Kinect to review CAD models in the VR system. The design review interface shown in Figure 3-12 displays the operation menu, virtual mouse, and product model. The menu has five review commands placed on the top of the interface window. The product model and component detail are located on the left top of the window. The white hand in the blue check box is a virtual mouse. The product model is placed in the middle of the interface. Using the interface, the user can change the view via body gestures. The gestures are matched based on the distance and angular thresholds among skeletal joints. The virtual mouse is controlled by left and right hands in the space area, which is programmed using "win32.api". The virtual mouse can move in the interface window to perform functions of controlling checkboxes of the menu. The checkbox is used to avoid the gesture interference and execute commands correctly. If a checkbox is selected, the related body gestures will trigger corresponding commands to distinguish common gestures.

The application is coded using Python programming language in Vizard for a full access to open-source community libraries and toolkits. The body gesture recognition consists of three processes, the body joint capturing, data transmission and gesture matching. The body joint capturing uses FAAST with the support of OpenNI and Microsoft SDK libraries.

The OpenNI framework is an application programming interface that provides a middleware component to retrieve the images from Kinect and to determine the user limbs positions. The data transmission is based on VRPN (Virtual-Reality Peripheral Network) in FAAST with a set of classes and servers that are designed for data transmission in a network-transparent interface in VR applications (*VRPN*, 2016). The gesture matching is conducted in the Vizard VR system. The process consists of four steps: 1) software initialization, 2) importing models, 3) starting VRPN and adding sensors, and 4) defining body gesture triggered functions. The thresholds for different gestures are set in Step 4, the distance threshold between sensor0 and sensor14 is between 0.01 and 0.2 meters. The sample code of these four steps is shown in Figure 3-13.

The design review of a 168CC kart is conducted for the application of the proposed system. Solidworks models of the 168CC kart are imported into Vizard in separate parts: the frame, bumper, left front wheel, left rear wheel, right front wheel, right rear wheel, seat and accelerator, steering, motor and electric control. The part models are shown in Figure 3-14, a complete kart model is shown in Figure 3-15.

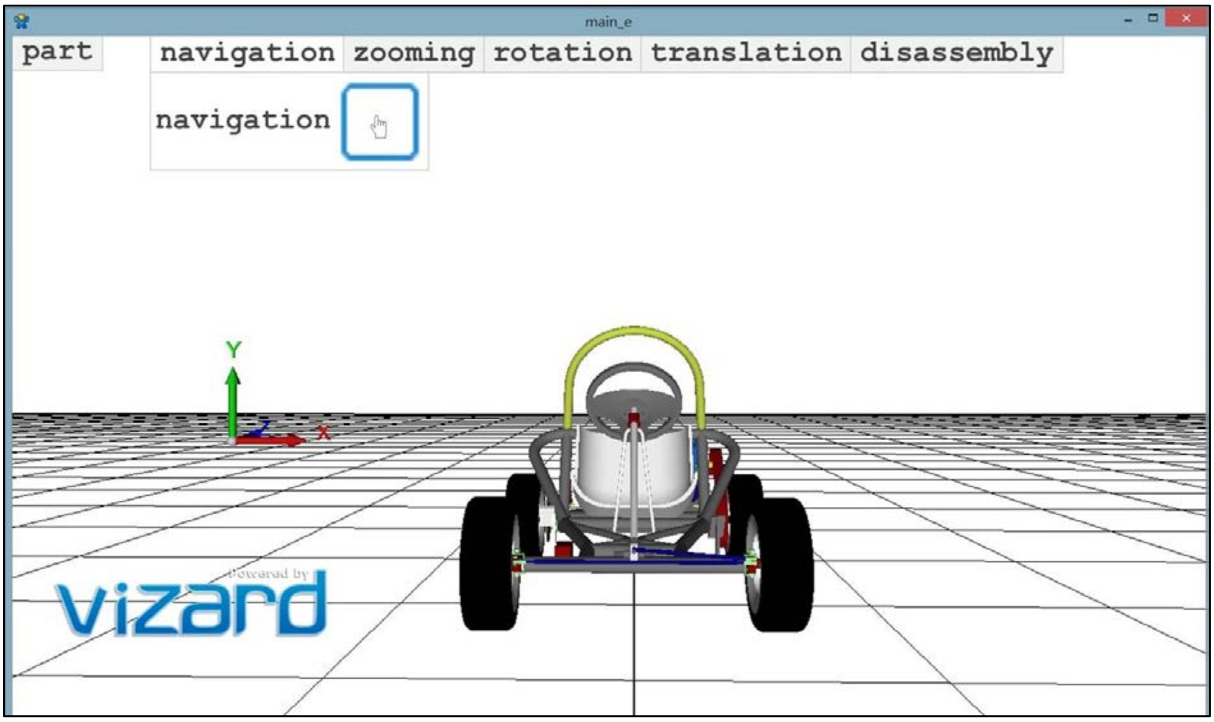


Figure 3-12 Design review interface

Programming code

```
#Software initialization
import viz
import vizshape
import vizact
import vizmat
import vizproximity

viz.go()
grid = vizshape.addGrid()

#Importing models
bumper = viz.addChild('bumper.osgb',pos=(0,0.2,1.5))

#Starting VRPN and adding sensors
trackers = []
vrpn = viz.addExtension('vrpn7.dle')
t0=vrpn.addTracker('Tracker0@localhost', HEAD)
t14=vrpn.addTracker('Tracker0@localhost', RIGHTHAND)
trackers.append(t0)
trackers.append(t14)

#Defining body gesture triggered functions
spin1=vizact.spin(0,1,0,10,2)

def spin_1():
    global enterevent1
    if enterevent1==True:
        bumper.runAction(spin1,pool=1)

def spin_11():
    t0pos=t0.getPosition()
    t14pos=t14.getPosition()
    distance014=vizmat.Distance(t0pos,t14pos)
    print ('distance014=', distance014)
    if 0.01<distance014<0.2:
        spin_1()
vizact.ontimer(1,spin_11)
```

Figure 3-13 Coding example

In the design review process, users navigate in the virtual environment to review product models. When a specific model is selected in the menu, scaling, rotation and translation commands can be applied to the model. Users can have a close look at the model from different viewpoints. Figure 3-16a shows a general view of the disassembled model. The initial positions of individual models are predefined in the VR system. With the frame located in the center, four wheels scatter around the frame and motor is located at the right side of the frame. User can navigate around models. Figure 3-16b is a translation scene. The selected model (left front wheel) is translating its position in the interface. As the translation gesture and navigation gesture share the same features, either the translation gesture or navigation gesture will be activated during the design review process. The activation and deactivation process is achieved using the virtual mouse to operate the menu. Figure 3-16c is a rotation scene. A rotation command (around X axis) is applied to the selected model (motor and electric control). Figure 3-16d is a scaling scene of the selected model in Figure 3-16c. During the review process, users can change the positions of the main view in the virtual scene by acting navigation gestures, the whole scene can be observed from different views.

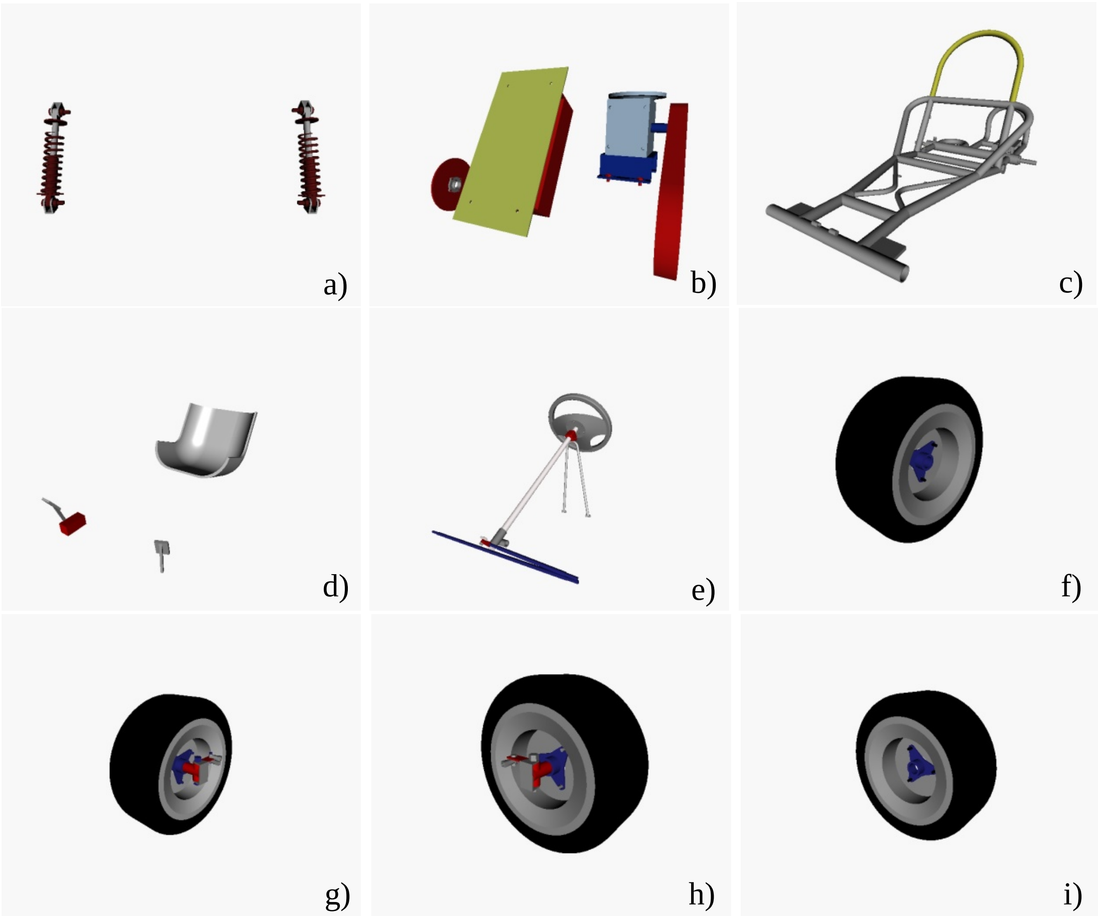


Figure 3-14 Parts model

Figure 3-15 Kart model

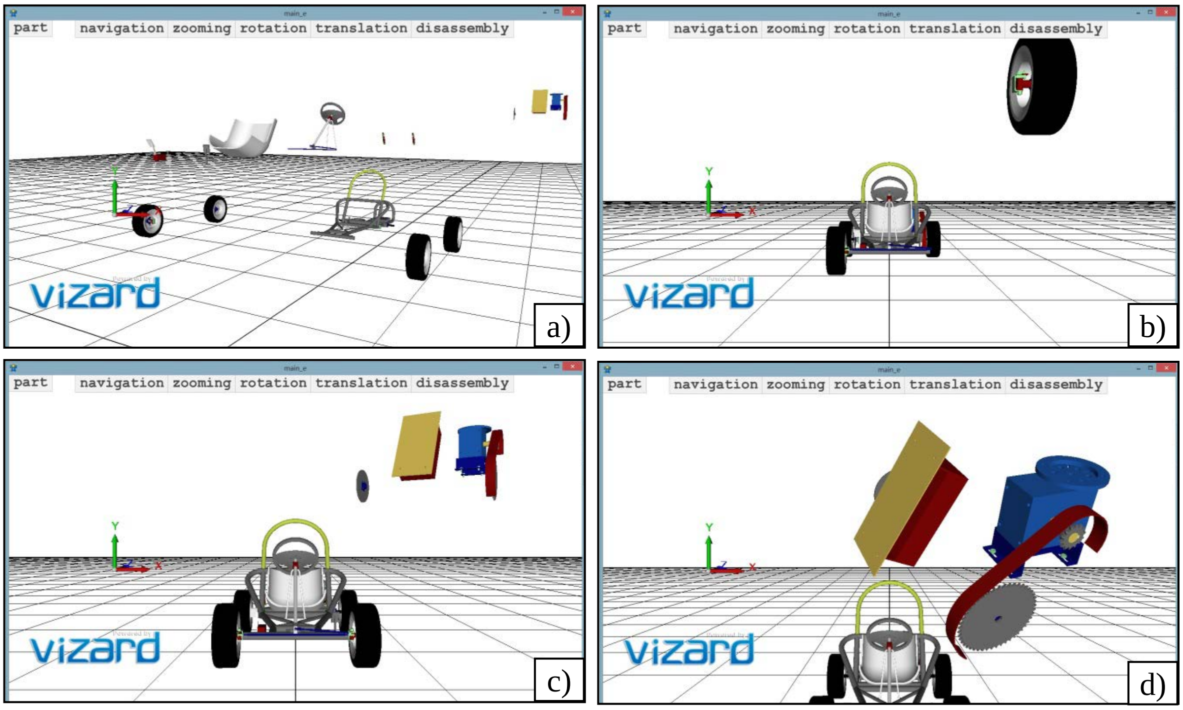


Figure 3-16 Design review

3.4. System evaluation

Experiments were conducted to evaluate the system. Four methods were compared to select the best way to control the virtual mouse in order to reduce errors caused by data floating. Gesture recognition experiment was carried out to evaluate the gesture recognition rate and a user test was conducted.

3.4.1. Virtual mouse control

The captured Kinect skeletal data are unstable to control the virtual mouse. Four methods are applied, Gaussian filter, average, weighted average#1, and weighted average#2, to smooth data transmitted from the Kinect sensor.

For the Gaussian filter, the latest 500 samples are collected for smoothing and one-dimensional Gaussian formula is shown in Eqn. (3-6) where σ represents the standard deviation for the

Gaussian kernel. The bigger σ is, the smoother the Gaussian distribution is. An example of the Gaussian filter calculation process is described in Figure 3-17. An example of the processed data is shown in Figure 3-18. The red line in the diagram is the smoothed data and the black line with the obvious fluctuation is the original data.

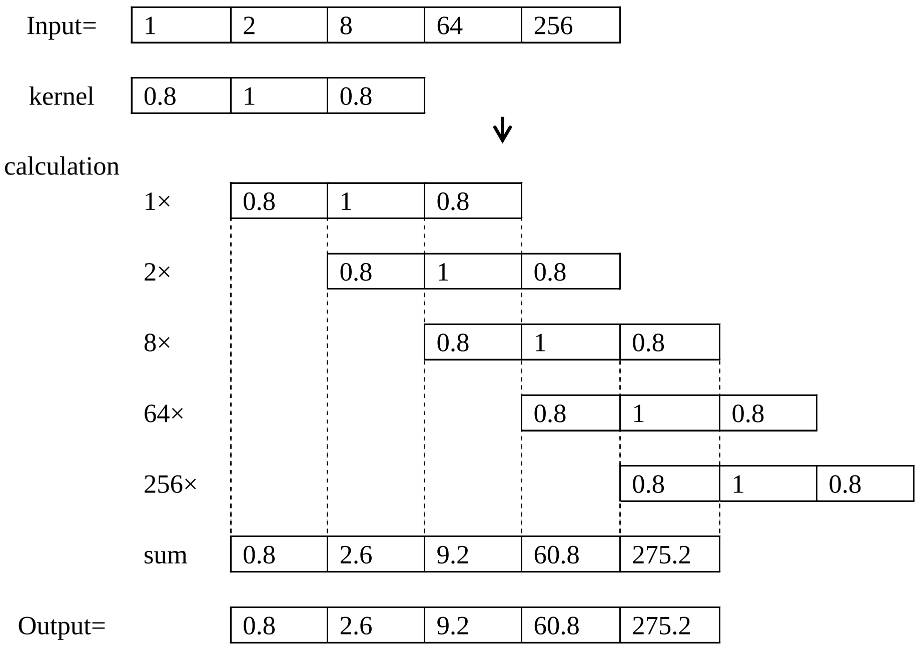


Figure 3-17 Gaussian filter calculation

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} \quad (3\text{-}6)$$

For the average method expressed in Eqn. (3-7) where d1 is the latest one, d2 the second latest, d3 third latest, d4 forth latest and d5fifth latest, the latest 5 data are collected from the real-time updating data set to calculate the average.

For the weighted average#1 expressed in Eqn. (3-8), the latest 5 data are collected to calculate the weighted average using weights of 1, 2, 3, 4, 5 from the earliest one to latest one, respectively.

For the weighted average#2 expressed in Eqn. (3-9), the latest 5 data are used to calculate the weighted average with weights of 5, 4, 3, 2, 1 from the earliest one to latest one.

$$A=\left(d1+d2+d3+d4+d5\right)/5 \qquad (3\text{-}7)$$

$$WA1=\left(d1*1+d2*2+d3*3+d4*4+d5*5\right)/15 \qquad (3\text{-}8)$$

$$WA2=\left(d1*5+d2*4+d3*3+d4*2+d5*1\right)/15 \qquad (3\text{-}9)$$

To conduct the experiment, three users were asked to perform a task of selecting the required command from the menu. In the experiment, the task is to select the rotation command around x axis. For each smoothing method, they were asked to perform the task for 5 times. Time of completing the task was recorded. The results of these four methods were compared to get the best one. The results are shown in Table 3-2.

It is noticed that using original data for the virtual mouse to complete the task takes the most time. It takes less time to complete the task using smoothed data. The method of using the Gaussian filter takes the least time. Therefore, the Gaussian filter is applied to smooth the data in order to improve the stability and effectiveness of the virtual mouse control.

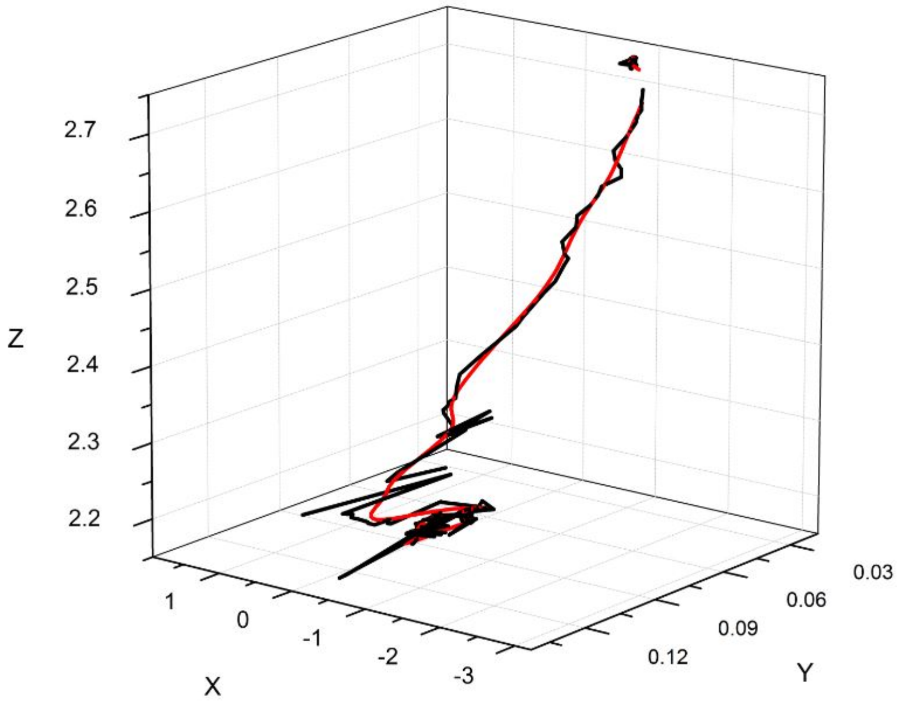


Figure 3-18 Comparison between original data and Gaussian filtered data

Table 3-2 Comparison of the average completion time

Method	AFT User1	AFT User2	AFT User3	AFT
Original	10.41s	10.27s	10.49s	10.39s
Gaussian	5.54s	5.60s	5.57s	5.57s
Average	5.74s	5.90s	6.18s	5.94s
Weighted average #1	6.38s	6.53s	6.32s	6.41s
Weighted average #2	6.39s	6.64s	6.50s	6.51s

AFT: Average finishing time

3.4.2. Gesture recognition rate

Twelve gestures are tested in the gesture recognition to evaluate the robustness of the proposed system. Three users were taught to perform the gestures before the test. They perform each gesture for 30 times with a pause of 2-3 seconds each time. The success of the triggered command was recorded to calculate the recognition rate. The average recognition rate of the three users is also calculated. The result is shown in Table 3-3.

Recognition rates of the designed body gestures are over 80% from Table 3-3. The recognition rate of static gestures is higher than dynamic gestures. However, the evaluation only required users to perform one command each time and the number of trials is limited.

Table 3-3 Gesture recognition rate

Gesture	Type	RR User1	RR User2	RR User3	ARR
Move forward	Static gesture	83.3%	90.0%	86.7%	86.7%
Move backward	Static gesture	86.7%	86.7%	90%	87.8%
Move upward	Static gesture	90%	86.7%	90%	88.9%
Move downward	Static gesture	86.7%	86.7%	90%%	87.8%
Turn right	Static gesture	93.3%	90.0%	90.0%	91.1%
Turn left	Static gesture	93.3%	93.3%	90.0%	92.2%
Scaling up	Dynamic gesture	86.7%	83.3%	83.3%	84.3%
Scaling down	Dynamic gesture	86.7%	83.3%	80.0%	83.3%
Rotate (x axis)	Dynamic gesture	83.3%	83.3%	80.0%	82.2%
Rotate (y axis)	Dynamic gesture	83.3%	83.3%	83.3%	83.3%
Rotate (z axis)	Dynamic gesture	83.3%	83.3%	86.7%	84.4%
Explode	Dynamic gesture	86.7%	83.3%	86.7%	85.6%
Assembly	Dynamic gesture	83.3%	86.7%	83.3%	85.6%

RR: recognition rate; ARR: average recognition rate

3.4.3. User test

A user test was designed to review product models in Solidworks and in the proposed system separately to compare the solutions. A questionnaire survey was conducted to rate these two systems. There were 10 student users taking part in the test. All the participants had used CAD software before and were familiar with CAD commands. But none of them had experience in the gesture-based interaction. Before the reviewing process in the two systems, a training process of the gesture-based system was introduced before the test. During the training process, they were taught to perform gestures to review the models and they can ask question about the gesture operations in order to learn and remember gestures. After the model review in both systems, users were asked to evaluate the ease of learning and understanding gestures, naturalness and intuitiveness of the both systems. For the learning and understanding (LU), the scale is set from 1 to 5 (1 represents “easy”, 3 represents “moderate”, 5 represents “difficult”). For the naturalness (N) and intuitiveness (I), the scale is set as 1 to 5 (1 represents “awkward”, 2 represents “boring”, 3 represents “acceptable”, 4 represents “good”, 5 represents “perfect”). As the sample size (participants) is small, non-parametric method (Mann-Whitney) was applied to compare the statistically significant differences of the scale value in the questionnaire (Guerra, Gidel and Vezzetti, 2016). The results are shown in Table 3-4.

Though the average scale of LU using gestures is slightly lower than the keyboard and mouse, the Mann-Whitney U for LU is larger than 0.05 and it means that there are not significant differences in the feedback of users comparing LU. Therefore, learning and understanding difficulty of gestures and the keyboard and mouse for CAD operations is basically the same. The Mann-Whitney U for N and I is smaller than 0.05 and it means that there are differences comparing naturalness and intuitiveness of using the mouse and keyboard-based CAD system and the gesture-based system for the design review. Comparing the mode and mean of the users' feedback, most users responded that it was “acceptable” to use the keyboard and mouse for interactions. For the gesture-based design review system, the majority of users thought it “good” to review the models in such a system. In general, difficulties of learning and understanding the designed gestures and the keyboard and mouse-based commands for CAD operation are similar. The naturalness and intuitiveness are improved using the gesture-based design review system.

Table 3-4 Statistical results

Likert-type	Median	Mode	Mean	Mann-Whitney
LU G	2	2	1.6	0.3724
LU K	2	2	1.9	
N G	4	4	4.2	0.0007
N K	3	3	2.7	
I G	4	4	4.0	0.0076
I K	3	3	3.3	

This chapter presented a body gesture-based design review interface. Body gestures are applied to manipulate CAD models in a VR system. Product assembly and modular disassembly are simulated for users to not only review the assembly model but also access to the product modules. However, the detail parts in modules are not able to review. The following chapter will introduce hand gestures for the product detail review.

Chapter 4
Hand gesture-based interface for design review

4.1. Proposed system

A hand gesture-based design interface is developed using the Vizard VR software (Vizard, 2015) and Leap Motion Controller (LMC). There is no study done combining Vizard and LMC for the design review. User's hand gestures are applied for user interactions with 3D models in the VR environment to improve the naturalness and intuitiveness of HCI. Figure 4-1 shows the structure of the proposed system. The CAD system is used to design product for the model manipulation in the VR system. The VR system, connecting with the LMC application programming interface (API) using Python programming, provides a platform for user interactions to CAD models using gestures. Leap Motion Python SDK is used for capturing hand data, which is shown in Figure 4-2. Figure 4-2a shows left hand data and Figure 4-2b shows right hand data. Captured data from the LMC are mapped with gesture templates to obtain the user's command to operate the model.

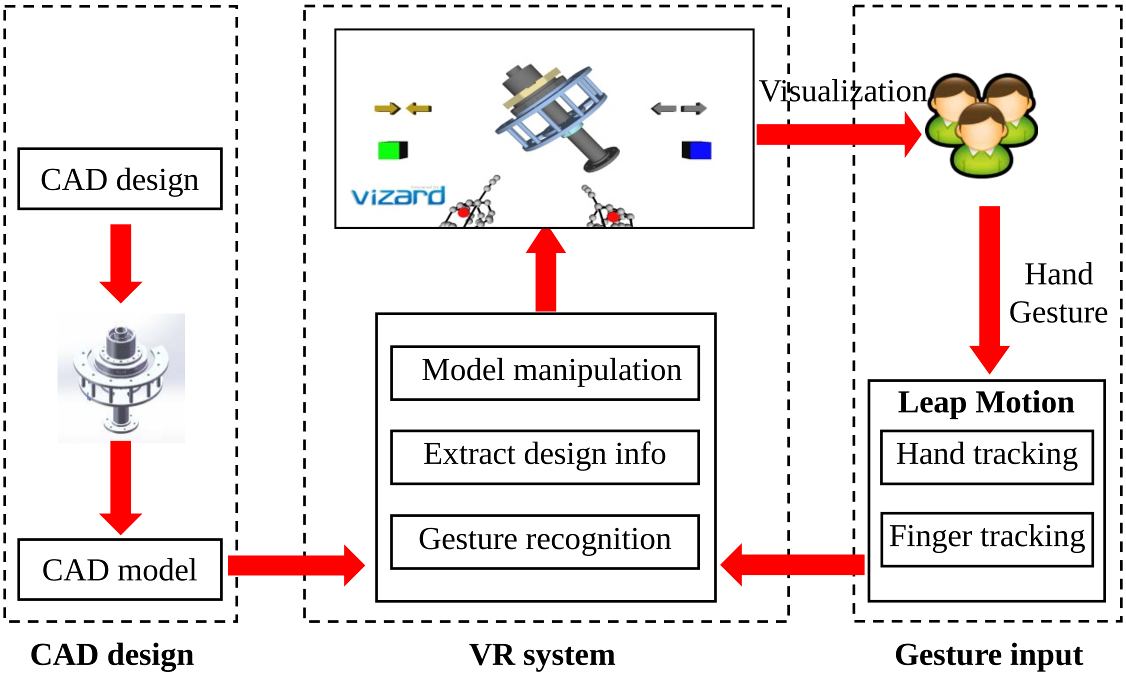


Figure 4-1 System structure

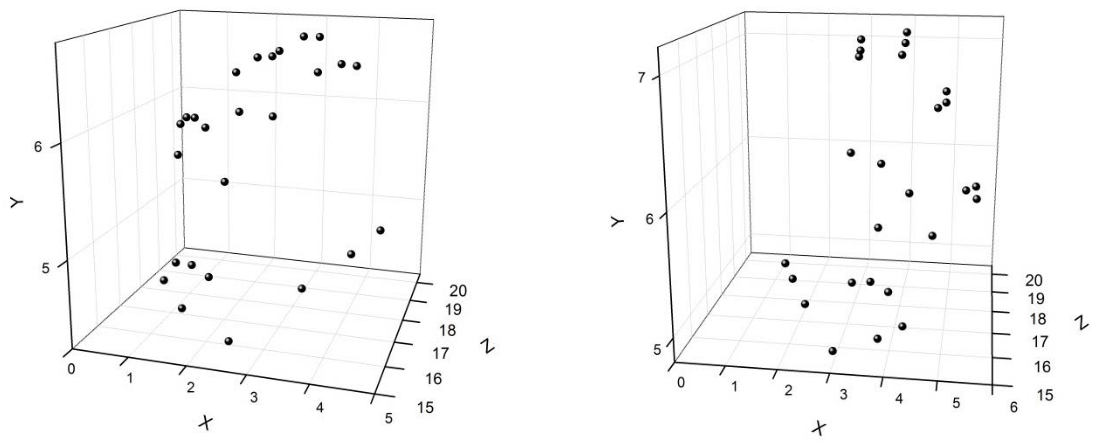


Fig. 4-2a

Fig. 4-2b

Figure 4-2 LMC hand data

4.2. Gesture and mapping

4.2.1. Gesture design

Commands in the review mode of most CAD systems can be classified into translation, rotation and scaling operations (Song et al., 2014). These commands are mainly operated using the computer mouse and keyboard. Hand gestures are proposed in this research to replace the computer mouse and keyboard for the command input as shown in Table 4-1.

Users have to keep their arm floating in the air against gravity when gestures are used to manipulate the model. The number of gestures is minimized to reduce users' cognitive load for remembering hand positions and movements. Four simple gestures are proposed based on the user study in the previous research (Thakur and Rai, 2015). Gesture definitions in this research are shown in Table 4-2.

Attributes in Table 4-2 are used to define the gesture templates as $T = \{T_1, T_2, T_3\}$, where T_1 represents the translation gesture, T_2 represents the rotation gesture and T_3 represents the scaling gesture as shown in Table 4-3. Considering the hand size of different users, a threshold τ is used to match gestures from different users. The switching gesture in Table 2 is predefined in the LMC's application programming interface (API), which is recognized by the API and it is only mapped with the switching command in the VR System.

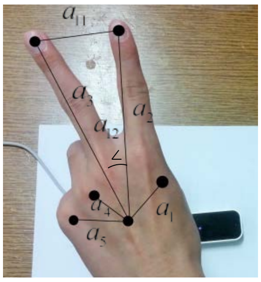
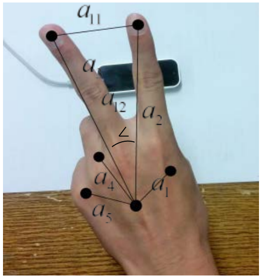
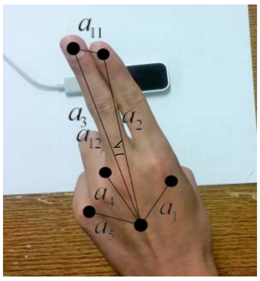
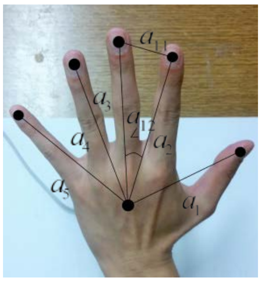
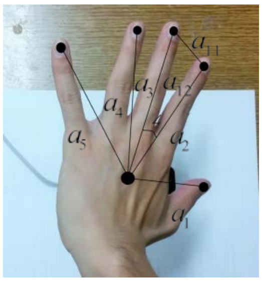
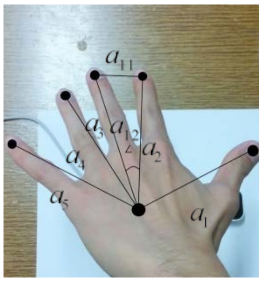
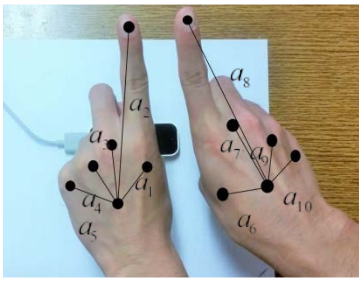
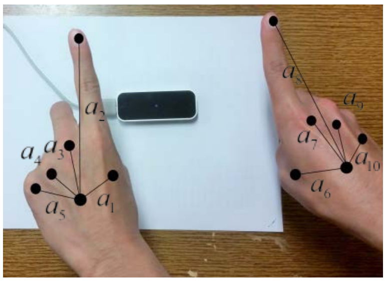
The translation operation is a pinch gesture using the left hand. The pinch gesture serves grabbing the model and translating the model in the interaction area (Thakur and Rai, 2015). The displacement from the pinch gesture is used as a parameter for the translation. The rotation operation uses a stretched left hand. The rotation operation is based on the Euler rotation in the VR system.

The scaling operation uses two index fingers moving apart from each other. The gesture acts as a trigger signal and adjust the zooming parameter. The user zooms in or out by widening or narrowing the distance between two index fingers.

Table 4-1 CAD commands and proposed gestures

Software	**CAD commands**	**Operations**	**Functions**	**Gestures**
Solidworks	Pan	Translation	Translate the model	Translation
AutoCAD	3D pan	Translation		
CATIA	Pan	Translation		
Solidworks	Rotate	Rotation	Rotate the model	Rotation
Solidworks	Roll	Rotation		
AutoCAD	Rotate	Rotation		
AutoCAD	3D Rotate	Rotation		
CATIA	Rotate	Rotation		
Solidworks	Zoom to fit	Scaling	Zoom in or out to see the entire or certain area of a model	Scaling
Solidworks	Zoom to area	Scaling		
Solidworks	Zoom in/out	Scaling		
AutoCAD	Scale	Scaling		
AutoCAD	3D zoom	Scaling		
CATIA	Fit all in	Scaling		
CATIA	Zoom area	Scaling		
CATIA	Zoom in/out	Scaling		
			Switch components of the input model	Switching

Table 4-2 Gesture definition

<table>
<tr><td colspan="3">Translation</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td colspan="3">Index and middle fingers of the left hand are used to perform pinch gestures for grab and release commands. When a model is grabbed, it can be translated to any place in the interaction window. The model is released by opening fingers. A template of the translation gesture is defined using joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), index and middle (a_{11}) and the angular relationship between index and middle fingers (a_{12}).</td></tr>
<tr><td colspan="3">Rotation</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td colspan="3">The model can be rotated based on the rotation of five fingers of the left hand. A template of the rotation gesture is defined using joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), index and middle (a_{11}) and the angular relationship between index and middle fingers (a_{12}).</td></tr>
<tr><td colspan="3">Scaling</td></tr>
<tr><td colspan="2"></td><td></td></tr>
<tr><td colspan="3">Two index fingers are used for scaling the model. A template of the scaling gesture is</td></tr>
</table>

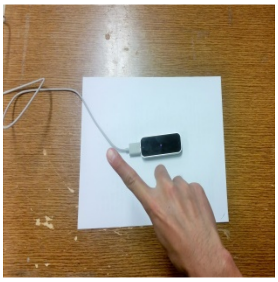
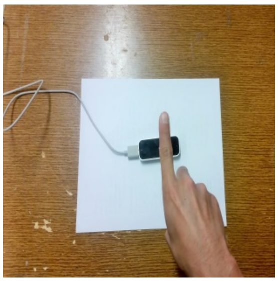
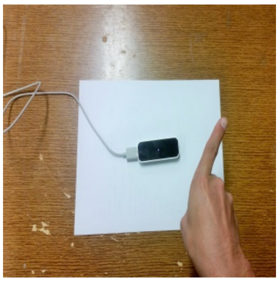
<table>
<tr><td colspan="3">defined using left hand joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), and right hand joints of palm and thumb (a_6), palm and index (a_7), palm and middle (a_8), palm and ring (a_9), palm and pinky (a_{10}).</td></tr>
<tr><td colspan="3">Switching</td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td colspan="3">Stretching index finger of the right hand and swiping it from left to right will change a model to the next model, swiping it from right to left will return to the original model.</td></tr>
</table>

Table 4-3 Template representation

A / T	a_1 (mm)	a_2 (mm)	a_3 (mm)	a_4 (mm)	a_5 (mm)	a_6 (mm)	a_7 (mm)	a_8 (mm)	a_9 (mm)	a_{10} (mm)	a_{11} (mm)	a_{12} (deg)
T_1	55	110	115	50	50	\	\	\	\	\	42	20
T_2	95	110	115	108	100	\	\	\	\	\	42	20
T_3	50	110	65	50	50	50	110	65	50	50	\	\
τ	15	15	15	15	15	15	15	15	15	15	10	8

A=attribute; T=template; τ=threshold value

4.2.2. Gesture mapping

The LMC captures the movement of hands and fingers with around 200 fps. Changes of the direction and displacement of hands can be identified by comparing data between two frames. Gesture mapping is to match the captured data to the template. A flow chart of the gesture mapping process is shown in Figure 4-3.

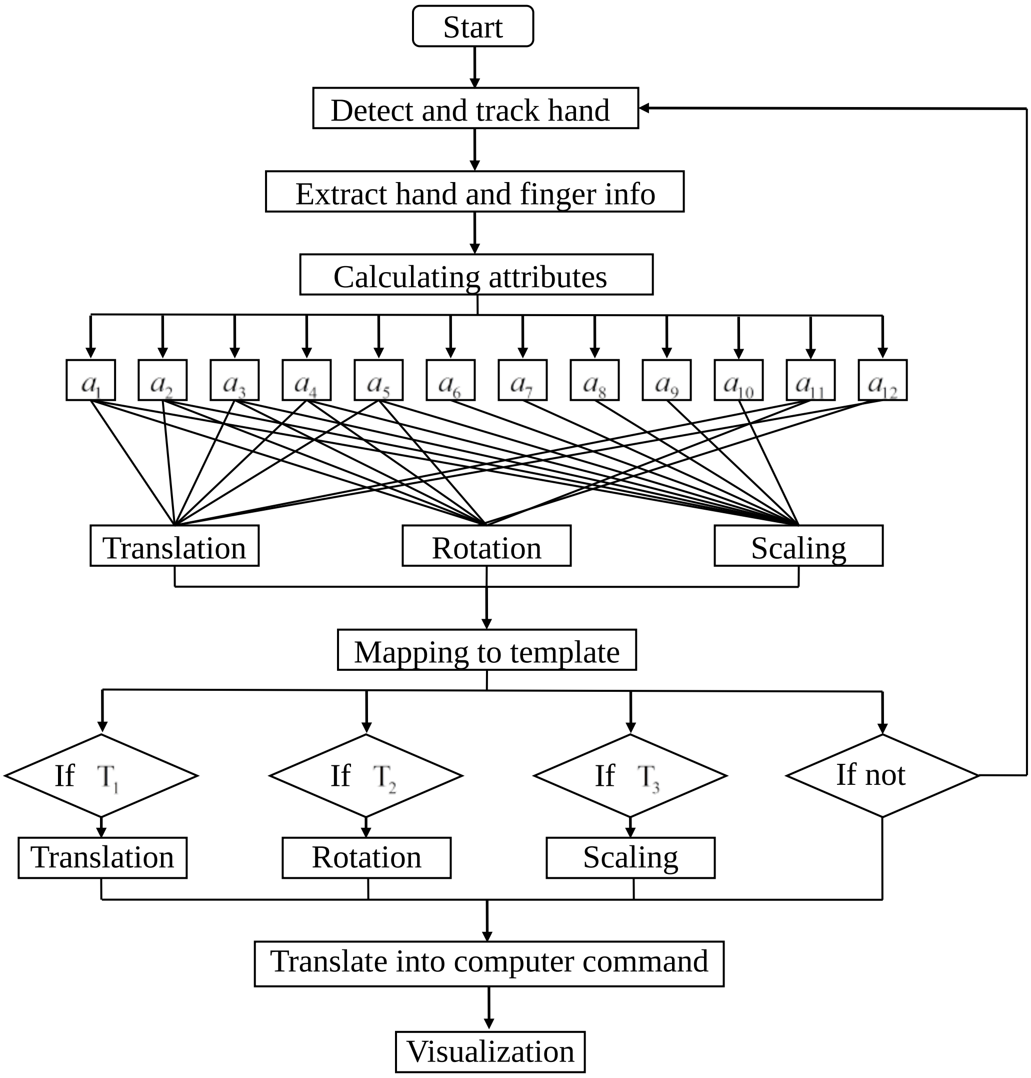


Figure 4-3 Gesture mapping flow chart

The LMC detects and tracks motions of hands. The captured data are calculated for values of attributes that are applied for template mapping. The combination of a_1, a_2, a_3, a_4, a_5, a_{11}, a_{12} is for translation and rotation gestures, the combination of a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10} is for the scaling gesture. If captured data are mapped with gesture templates, the corresponding commands will be triggered. Therefore, mapping the captured data to the template of gesture i is as follows:

$$g(T_i,\ D) = \begin{cases} 1, & \text{if } |a_b - d_b| < \tau_b\ (b = 0,1,2,\ldots,12) \\ 0, & \text{otherwise} \end{cases} \tag{4-1}$$

Where T_i represents a template for gesture i, a_b are attributes with a mapping threshold τ, d_b are calculated values of the attribute from input data. D is a distant vector calculated from the captured data. An example of the rotation gesture mapping is shown in Table 4-4. The LMC represents five fingers (thumb, index, middle, ring and pinky) with joints of the hand. The detail of representations of the rotation gesture is shown in Figure 4-4. As the result of $g(T_2,\ D)$ equals to 1, the rotation gesture is recognized.

Table 4-4 Gesture mapping example

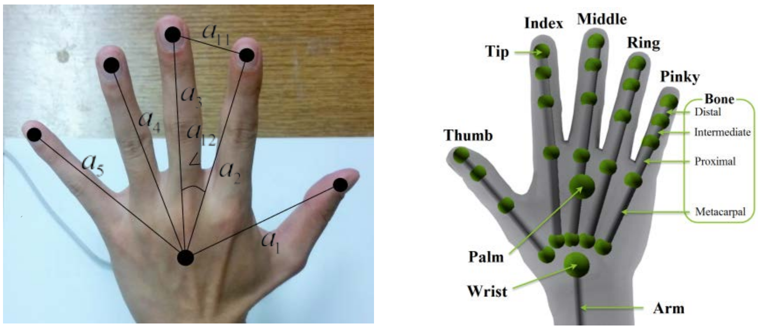
P / F	Input data (mm)		
	X	Y	Z
palm	-51.5	179.7	80.5
thumb	54.2	170.8	104.5
index	7.4	223.9	-5.2
middle	-27.9	232.8	-22.6
ring	-69.4	225.1	-20.4
pinky	-122.8	205.7	8.2

P=position; F=finger

➡

O / A	D	T_2	τ
a_1	108.8	95	15
a_2	113.0	110	15
a_3	118.3	115	15
a_4	112.0	108	15
a_5	104.8	100	15
a_{11}	40.3	42	10
a_{12}	19.92	20	8

O=output; A=attribute



Gesture mapping Leap motion controller hand bone

Figure 4-4 Example of the gesture mapping (*Leap Motion*, 2015)

4.3. System implementation and application

The interface is written using the Python programming language in the Vizard system. The LMC captures user's hand and finger data frame by frame for the gesture recognition. The recognized gestures are converted into CAD commands for the model manipulation. A rotating shaft model with eight parts and a vehicle wheel model with seven parts are applied as examples for the design manipulation.

The design interface consists of two parts: virtual hands and the user menu. Virtual hands, designed in the Vizard system, interact with models based on operations selected by the user from the menu. The menu consists of four selection buttons to perform different functions activated by touching buttons with a virtual hand. As shown in Figure 4-5, opposite arrows in the left corner of the window perform the assembly function in the interaction area of the interface. In this example, the position of the long shaft is fixed. Users can assemble the rest components with the translation gesture. When the distance between the selected component and long shaft reaches to a certain value, the selected components will be automatically assembled. Opposite arrows at the right corner perform the model disassembly. The green button on the left side activates the rotation and scaling commands in the manipulation process. The blue button on the right works as the deactivation of rotation and scaling commands. The manipulation process is shown in Figures 4-5 and 4-6, a-f show the manipulation of detailed models, g-i display the manipulation of the complete model.

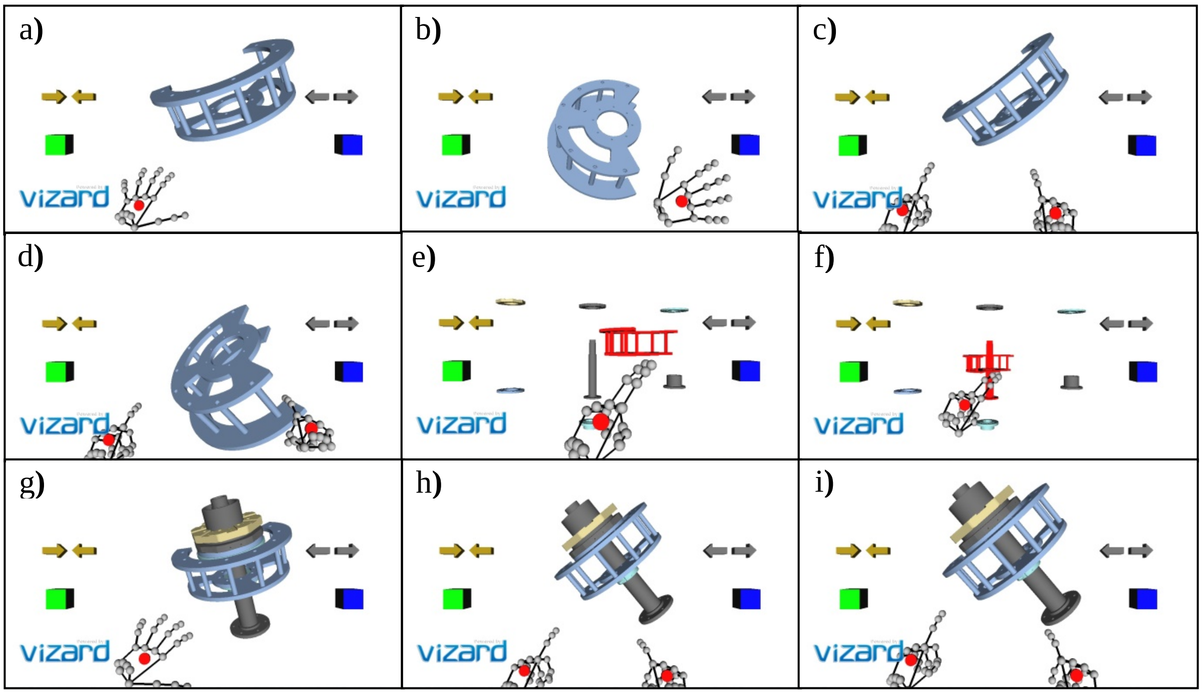


Figure 4-5 Design review interface-rotating shaft

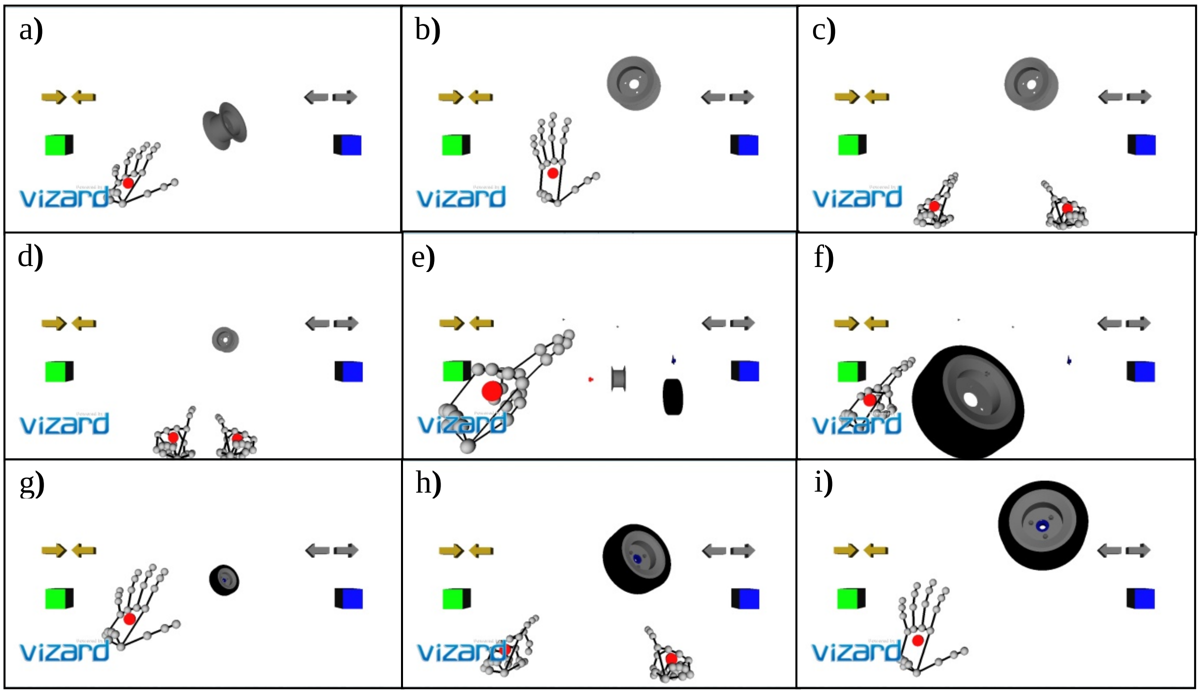


Figure 4-6 Design review interface-vehicle wheel

4.4. System evaluation

4.4.1. Gesture recognition rate

The gesture operations were tested by three users with different hand sizes for the gesture recognition to evaluate the robustness of the proposed system. Users perform each gesture for 50 times. The results are shown in Table 4-5. The recognition rates of the correct gestures are over 80%.

Table 4-5 Gesture recognition rate

Gesture	User	Total trial	Success	Recognition rate
Translation	User#1	50	43	86.0%
Rotation	User#1	50	43	86.0%
Scaling	User#1	50	45	90.0%
Translation	User#2	50	45	90.0%
Rotation	User#2	50	46	92.0%
Scaling	User#2	50	44	88.0%
Translation	User#3	50	44	88.0%
Rotation	User#3	50	43	86.0%
Scaling	User#3	50	46	92.0%

4.4.2. User test

Although the gestures can be effectively triggered, the detail performance of the gesture recognition is still unclear. A further user test was conducted as follows.

Fourteen students were invited to test the proposed system. Seven of the participants were familiar with CAD commands using the computer mouse and keyboard as input devices. Seven of the participants had no experience using the CAD software. None of fourteen students had used the gesture-based interaction system before. A two-step training process was introduced before the test. Firstly, they were taught to manipulate the design model in Solidworks using the mouse and keyboard as input devices. Secondly, they were taught to manipulate the design model in the VR system using hand gestures as input. After the training process, the participants can review the model by themselves in Solidworks and the VR system to compare the difference. A questionnaire was prepared for the comparison between gestures (G) and the mouse and keyboard (MK) as input methods for model manipulations in five aspects: learning time, intuitiveness, naturalness, cognitive load and ergonomic comfort. For the intuitiveness (I), naturalness (N) and ergonomic comfort (EC), the scale is set from 1 to 10 (1 represents “bad”, 10 represents “good”). For the learning time (LT), the scale is also set as 1 to 10 (1 represents “short”, 10 represents “long”). For cognitive load (CL), the scale is set as 1 to 10 (1 represents “low”, 10 represents “high”). Both parametric method (t-test) and non-parametric method (Mann-Whitney) were applied to compare the statistically significant differences of the scale value in the questionnaire (Guerra, Gidel and Vezzetti, 2016). The statistical results are shown in Table 4-6.

The P-value and Mann-Whitney U are all smaller than 0.05, which means that there are significant differences in the feedback of users comparing learning time, intuitiveness, naturalness, cognitive load and ergonomic comfort using hand gestures and the mouse and keyboard. From the mean, median and mode scales of the five aspects, it can be observed that the proposed gesture-based design interface is preferred for the design manipulation and the gestures are easy to learn and remember.

Table 4-6 Statistical results

Likert-type	**Mean**	**StDev**	**T-value**	**P-value**	**Median**	**Mode**	**Mann-Whitney U**
LT MK	6.43	1.50	5.20	0	6	5	0.0001
LT G	3.57	1.40			3	3	
I MK	5.79	0.80	6.68	0	6	6	0.0000
I G	8.07	1.00			8	8	
N MK	5.93	0.92	4.55	0	6	6	0.0002
N G	7.86	1.29			8	8	
CL MK	5.64	0.84	5.78	0	5	5	0.0000
CL G	3.36	1.22			3	3	
EC MK	5.71	0.99	6.61	0	6	6	0.0000
EC G	8.14	0.95			8	8	

This chapter presented a hand gesture-based design review interface using LMC. Hand gestures are applied to manipulate CAD models. Product module and detail parts are visualized and users can review the models using hand gestures. In the new chapter, the combination of body and hand gestures for assembly and detail design review system will be presented to make users have a fully understanding of product design.

Chapter 5

Integration of body and hand gestures for design review

5.1. System structure

In this section, a design review system combining Kinect for body gestures and LMC for hand gestures is developed. Both assembly design review and detail design review can be achieved in this system by selecting different review modes. A screenshot of the system is shown in Figure 5-1. The structure of the developed system is depicted in Figure 5-2. The system structure is similar to the design review system developed in Chapters 3 and 4. CAD design, gesture input and VR system are three main components of the developed system. The CAD design was described in Chapters 3 and 4. The gesture input combines both body and hand gestures by applying Kinect and LMC for different reviewing purposes. The body gestures captured by Kinect are used to review the assembly models and select menu. The hand gestures captured by LMC are used to review part details. The template mapping methods are applied for the gesture recognition in the Vizard VR system. The specific gesture design, system implementation and comparison of the proposed design review systems are discussed later in this Chapter.

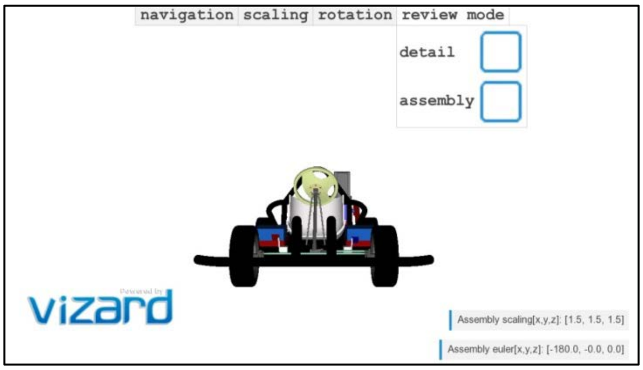


Figure 5-1 Design review interface using combined gestures

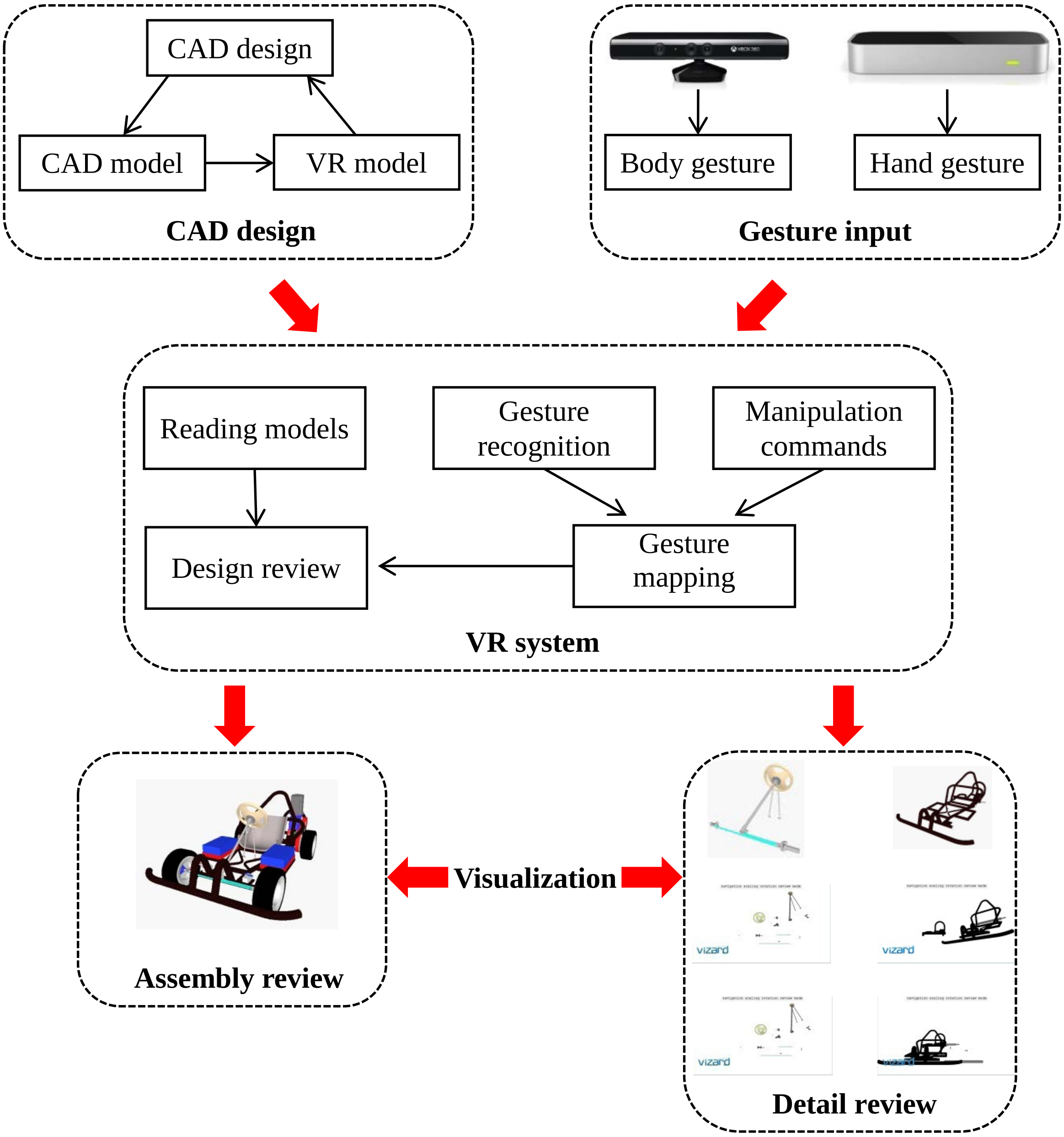
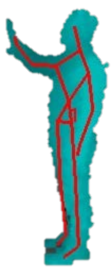


Figure 5-2 System structure

5.2. Gesture design and definition

There are two categories of gestures in the design review system: body gestures and hand gestures. The body gestures with six static gestures and two dynamic gestures are designed for the assembly review to allow user to manipulate the whole design model and navigate the model from different views. The hand gestures containing five dynamic gestures are designed for the detail review in which user can manipulate both the design model and detail parts. The gesture recognition method for body gestures and hand gestures was discussed in Chapters 3 and 4, respectively. The body gesture definition is shown in Table 5-1 and the hand gesture definition is in Table 5-2.

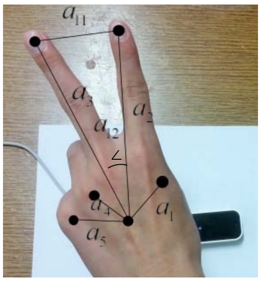
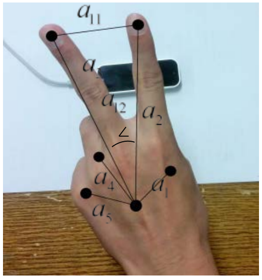
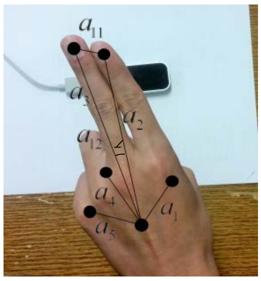
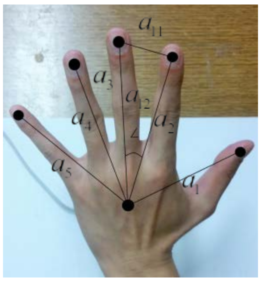
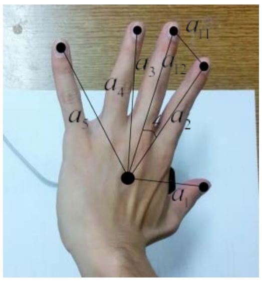
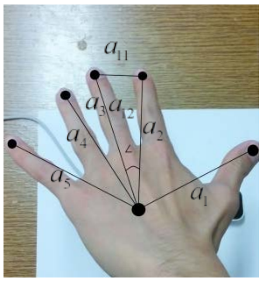
Table 5-1 Body gesture definition

Static Gesture		
 Front view	 Side view	Gesture: Move forward Category: Navigation When the right hand is raised forward and the arm is around 90 degree to the chest, the main view will be navigated forward.
 Front view	 Side view	Gesture: Move backward Category: Navigation When the left hand is raised forward and the arm is around 90 degree to the chest, the main view will be navigated backward.

<table>
<tr><td></td><td colspan="3">Gesture: Move upward
Category: Navigation
When the left hand is raised up and the elbow to the upper arm is around 90 degree, the main view will be navigated upward.—same as the move backward?</td></tr>
<tr><td></td><td colspan="3">Gesture: Move downward
Category: Navigation
When the left hand is raised down and the elbow to the upper arm is around 90 degree, the main view will be navigated downward.</td></tr>
<tr><td></td><td colspan="3">Gesture: Turn right
Category: Navigation
When the right hand is raised up by the side and the elbow to the arm is around 60 degree to the right leg, the main view will be turned right.</td></tr>
<tr><td></td><td colspan="3">Gesture: Turn left
Category: Navigation
When the left hand is raised up by the side and the elbow to the arm is around 60 degree to the right leg, the main view will be turned left.</td></tr>
<tr><td colspan="4">Dynamic Gesture</td></tr>
<tr><td>
Gesture#1</td><td>
Gesture#2</td><td>
Gesture#3</td><td>Gesture: Scaling up
Category: Scaling
The assembly model will be scaled up, when the user performs gesture#1, gesture#2 and gesture#3 in a sequence.</td></tr>
<tr><td>
Gesture#3</td><td>
Gesture#2</td><td>
Gesture#1</td><td>Gesture: Scaling down
Category: Scaling
The model will be scaled down, when the user performs gesture#3 first and followed by gesture#2, gesture#1.</td></tr>
</table>

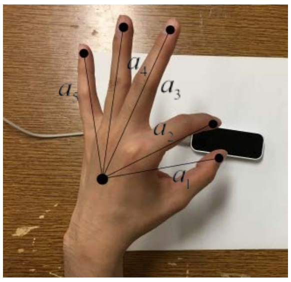
			Gesture: Rotate Category: Rotation The assembly model will be rotated in the x or y or z axis when user raises left and right hands around the neck and rotates the waist. When user rotates the waist to the left, the assembly model will rotate positively. When user rotates the waist to the right, the assembly model will rotate negatively.

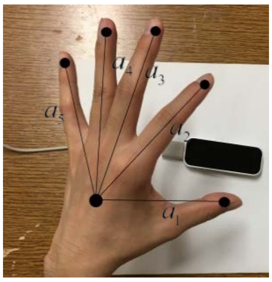
Table 5-2 Hand gesture definition

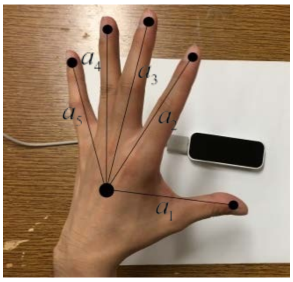
<table>
<tr><td colspan="3">Translation</td></tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
<tr><td colspan="3">Index and middle fingers of the left hand are used to perform pinch gestures for grab and release commands. When a model is grabbed, it can be translated to any place in the interaction window. The model is released by opening fingers. A template of the translation gesture is defined using joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), index and middle (a_{11}) and the angular relationship between index and middle fingers (a_{12}).</td></tr>
<tr><td colspan="3">Rotation</td></tr>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
<tr><td colspan="3">The model can be rotated based on the rotation of five fingers of the left hand. A template of the rotation gesture is defined using joints of palm and thumb (a_1), palm and index (a_2),</td></tr>
</table>

palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), index and middle (a_{11}) and the angular relationship between index and middle fingers (a_{12}).

Scaling

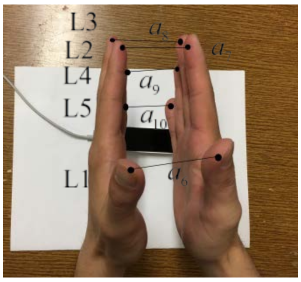


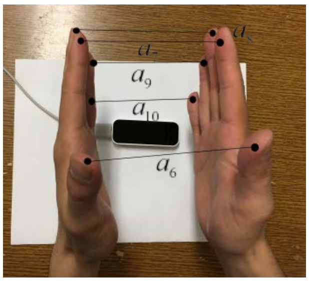


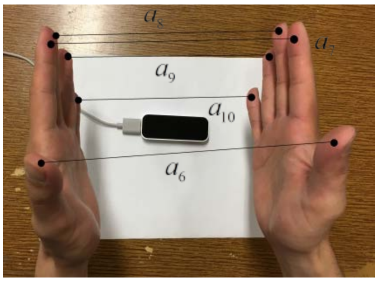


Left hand is used for scaling the model. A template of the scaling gesture is defined using left hand joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5).

Disassembly

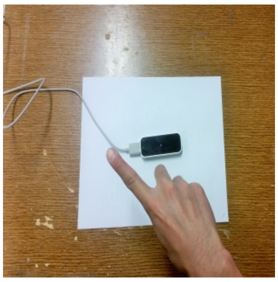
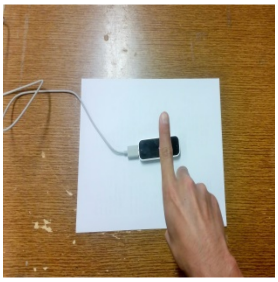
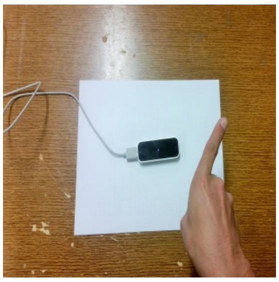






Two hands are used for disassembling the model. A template of the gesture is defined using angles between two hands' finger tips lines that L1 is the line between thumbs, L2 is indexes, L3 is middles, L4 is rings and L5 is pinkies. The included angle between L1 and L2 is a_6 , L2 and L3 is a_7 , L3 and L4 is a_8 , L4 and L5 is a_9 and L5 and L1 is a_{10} .

Switching

Stretching index finger of the right hand and swiping it from left to right will change a model to the next model, swiping it from right to left will return to the previous model.

5.3. Model transformation analysis

In the Vizard VR system, model transformations refer to the mapping from object to global coordinates. The model transformations such as translation, rotation and scaling are achieved using transformation matrices. The object transformation in the system can be represented by a 4×4 transformation matrix. The order of transformations applied is important as it will lead to different results if applied in the wrong order. In the Vizard system, transformations are applied to objects in the “reverse” order by following the OpenGL convention. It means that a new transformation matrix is needed to pre-multiply the existing one for the new transformation. For example, an object with a scale of 1.5×1.5×1.5 at the position of [1, 2, 3] can be represented in Vizard using a 4×4 matrix as Eqn. (5-1). A 60 degree rotation about y axis added to the object can be calculated using Eqn. (5-2). The object transformation can be represented using the outcome of Eqn. (5-2).

$$M=\begin{pmatrix} 1.5 & 0 & 0 & 0 \\ 0 & 1.5 & 0 & 0 \\ 0 & 0 & 1.5 & 0 \\ 1 & 2 & 3 & 1 \end{pmatrix} \tag{5-1}$$

$$M'=\begin{pmatrix} \cos 60 & 0 & -\sin 60 & 0 \\ 0 & 1 & 0 & 0 \\ \sin 60 & 0 & \cos 60 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1.5 & 0 & 0 & 0 \\ 0 & 1.5 & 0 & 0 \\ 0 & 0 & 1.5 & 0 \\ 1 & 2 & 3 & 1 \end{pmatrix}=\begin{pmatrix} 0.75 & 0 & -1.299 & 0 \\ 0 & 1.5 & 0 & 0 \\ 1.299 & 0 & 0.75 & 0 \\ 1 & 2 & 3 & 1 \end{pmatrix} \tag{5-2}$$

5.4. System implementation and applications

A kart model and a gearbox model with the three-layer structure are applied for the design review process. The kart model is shown in Figures 5-3 and 5-5a. The first layer is the assembly. The second module layer contains 13 modules: the accelerator and brake, battery, chain, electronics, frame, powertrain, seat, shock and suspension, steering, left front tire, right front tire, left rear tire and right rear tire. The third layer is detailed parts with around 130 components. The gearbox model is shown in Figures 5-4 and 5-5b. The design review system proposed in Chapter 3 includes the first layer and second layer for the assembly design review. The system in Chapter 4 works for the second and third layers in the detail review.

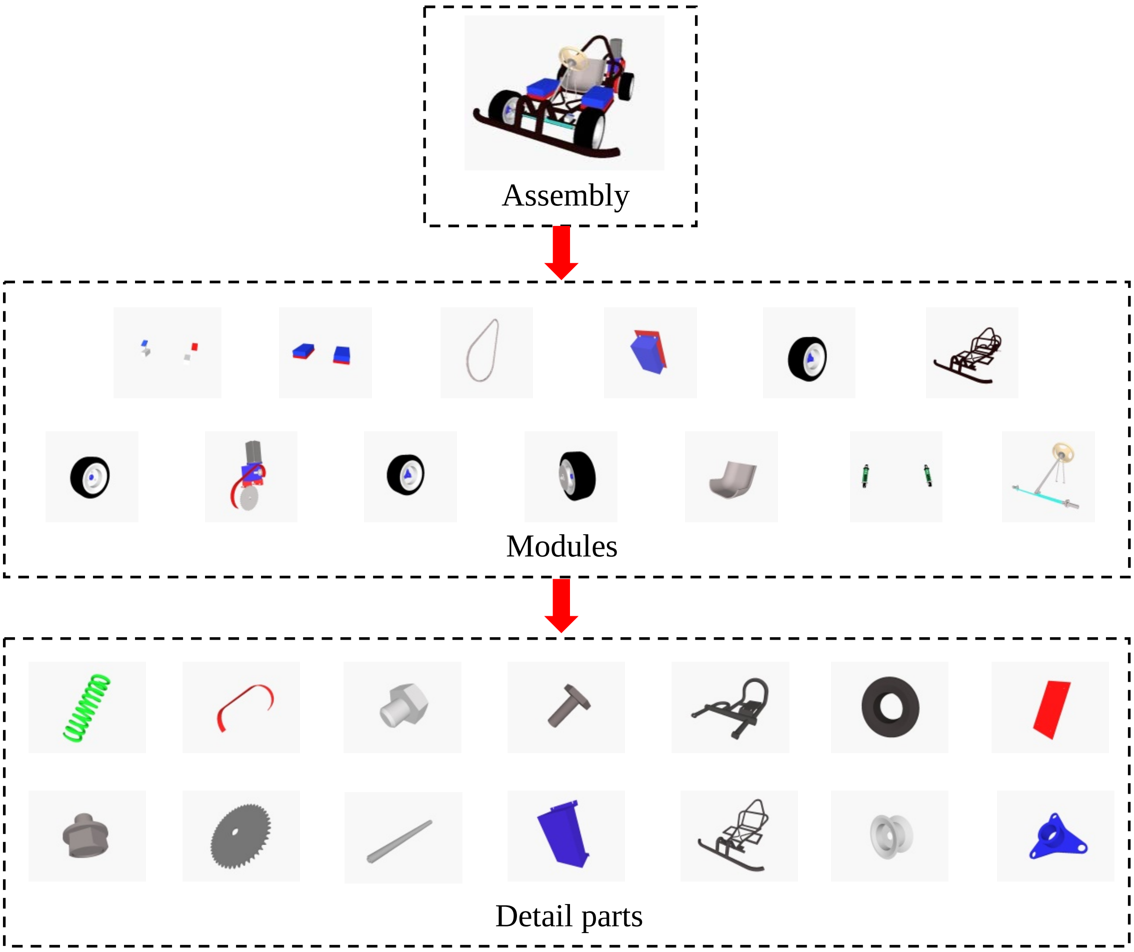


Figure 5-3 Design model of the kart

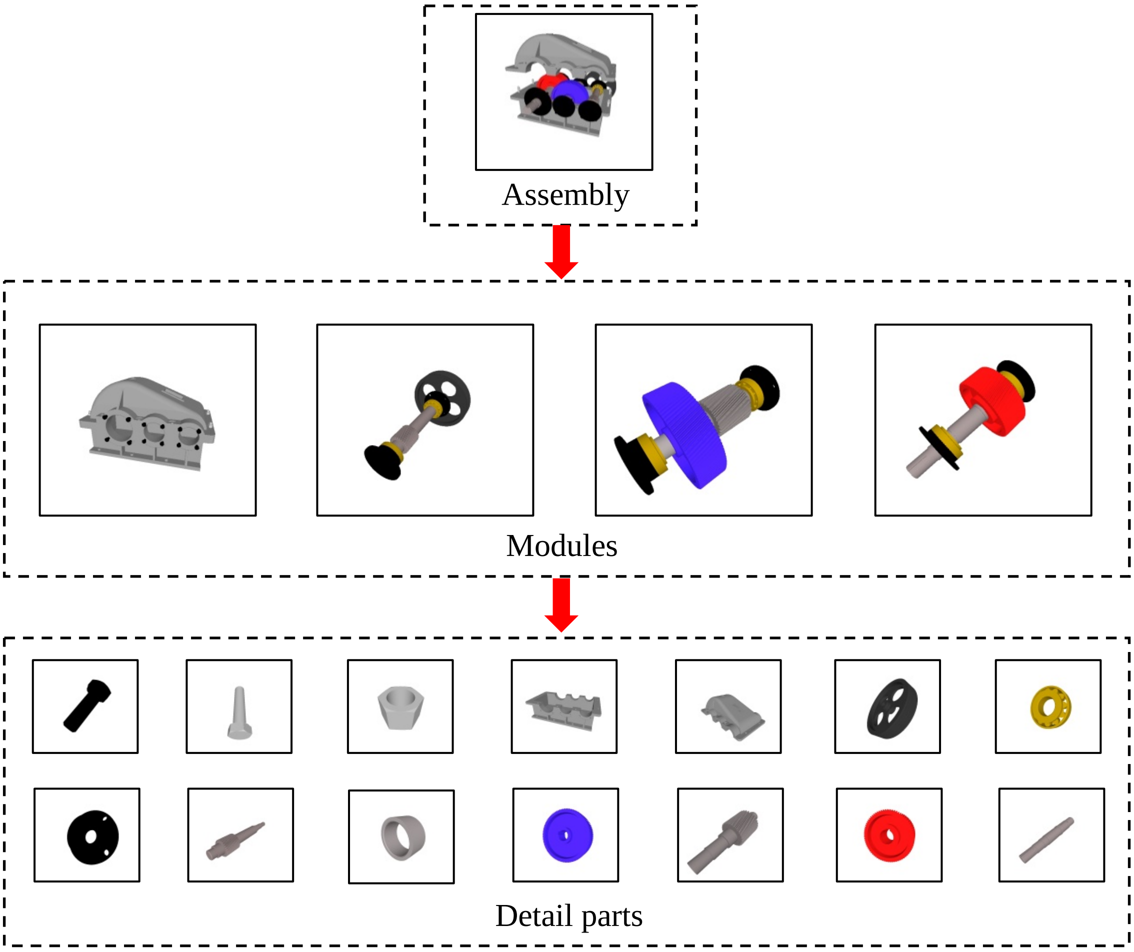


Figure 5-4 Design model of the gearbox

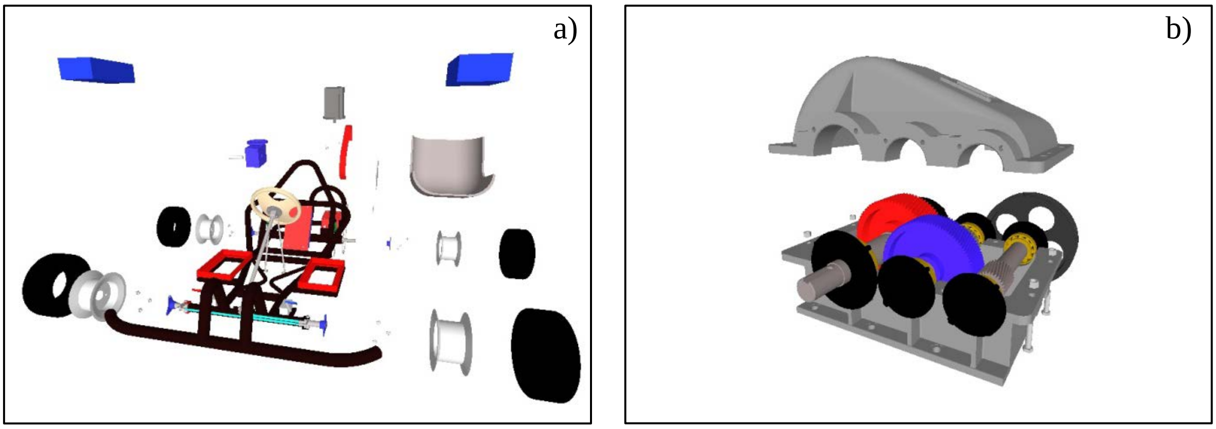


Figure 5-5 Constructions of design review models

In this Chapter, these three layers are combined for users to review product design in the assembly, modules and detail components.

The review interface shows the operation menu, virtual mouse, product model and a human skeleton in Figure 5-7a. The menu has three review commands and one switching mode checkbox placed on the top of the interface window. The virtual mouse is controlled by moving the left hand in front of the Kinect sensor within 1.7 meters. The assembly product model is initially located at the center of the window. The human skeleton on the right top of the window works as a reminder to indicate performing body gestures. When the body gesture is recognized, the skeleton will turn red and the operation command will be triggered.

The detail design review interface shows the hand model and product model in Figure 5-6b. The hand model indicates performing hand gestures.

As skeletal data are past to the VR software through the middleware FAAST for the body gesture recognition and hand data are transmitted through LMC SDK for the hand gesture recognition, these two data may cause interference during the design review process. A design review mode switch checkbox (Figure 5-6) is designed to avoid the interference. Before the detail review and assembly review process, users need to select the corresponding checkbox using the virtual mouse, and the relevant design review mode will be activated.

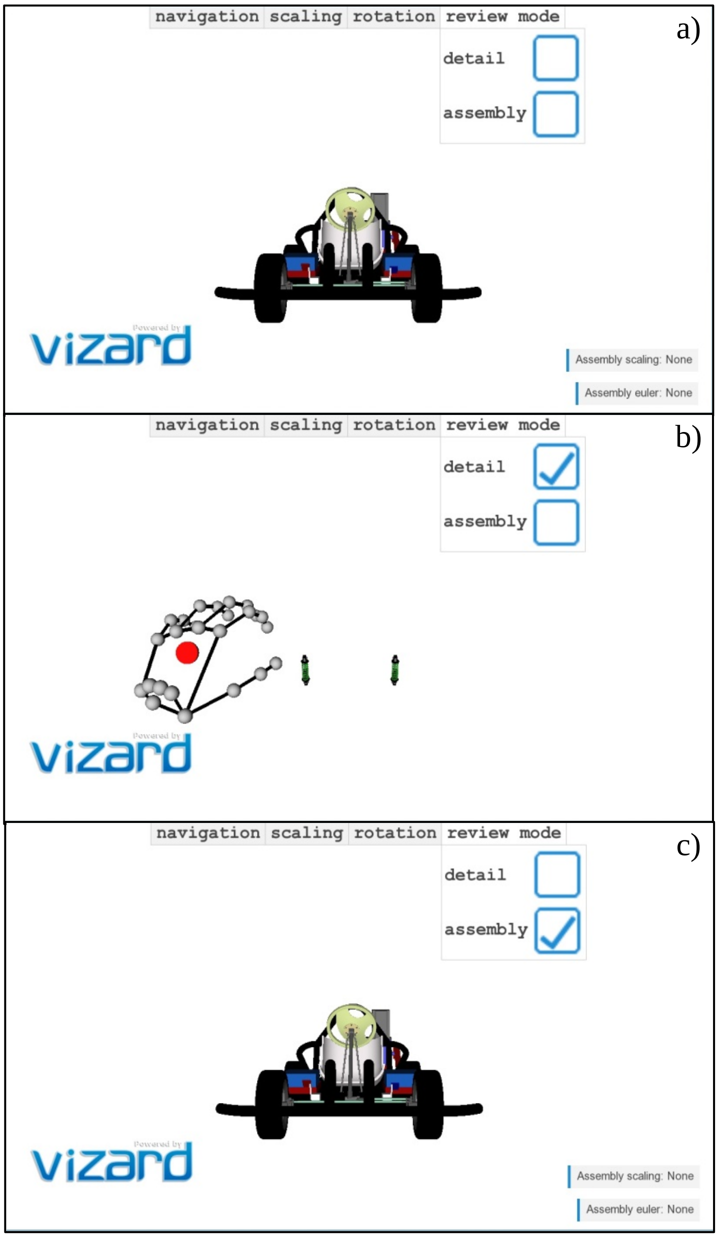


Figure 5-6 Review mode switching

5.4.1. Kart model design review

The assembly design review of kart model is shown in Figures 5-7, 5-8 and 5-9. Users navigate in the virtual environment from different viewpoints to review the assembly product model. Figure 5-7 shows the navigation operation of the assembly review. Figure 5-7a presents the assembly model and front view of the model. Figure 5-7b shows the left side view from the top after the user moved the scene upward. Figure 5-7c shows the right side view. Figure 5-7d presents the side view of the kart model.

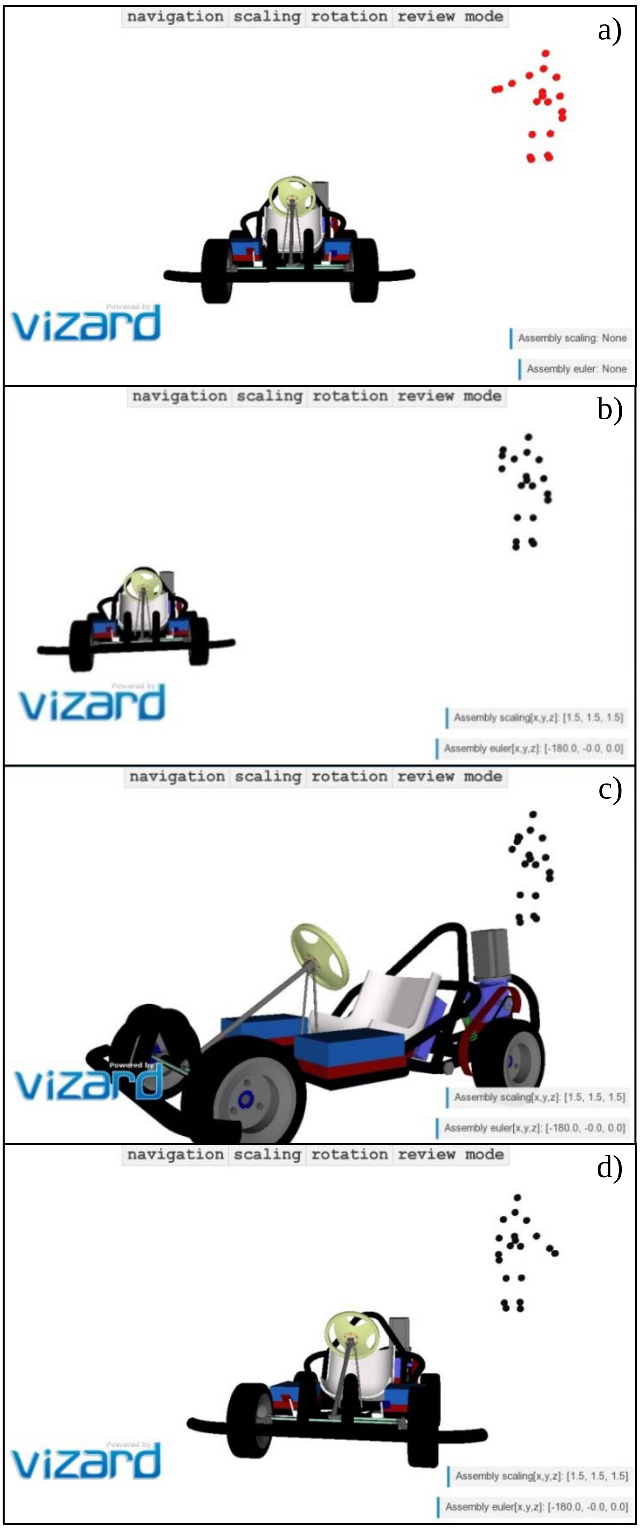


Figure 5-7 Assembly review navigation of kart

Figure 5-8 presents the scaling operation to scale up and down the kart model. Figures 5-8a and 5-8b present the assembly model after performing scaling up and down operations, respectively. Figure 5-8c shows the scaling up operation from a different viewpoint.

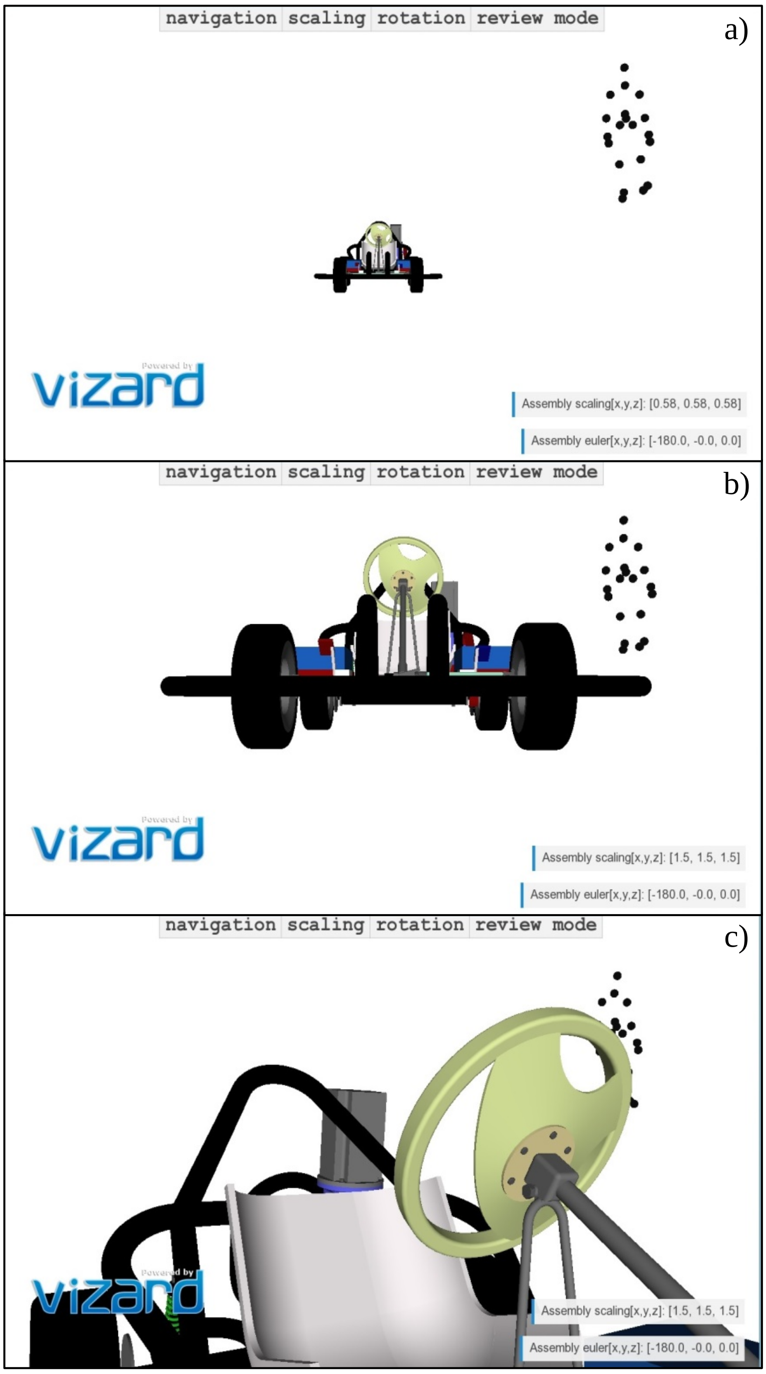


Figure 5-8 Assembly review scaling of kart

Figure 5-9 exhibits the rotation operation for the model to be rotated from x, y, z axes applying the rotation gesture. Figure 5-9a shows the z axis rotation, Figures 5-9b and 5-9c are combinations of z and y axes rotations.

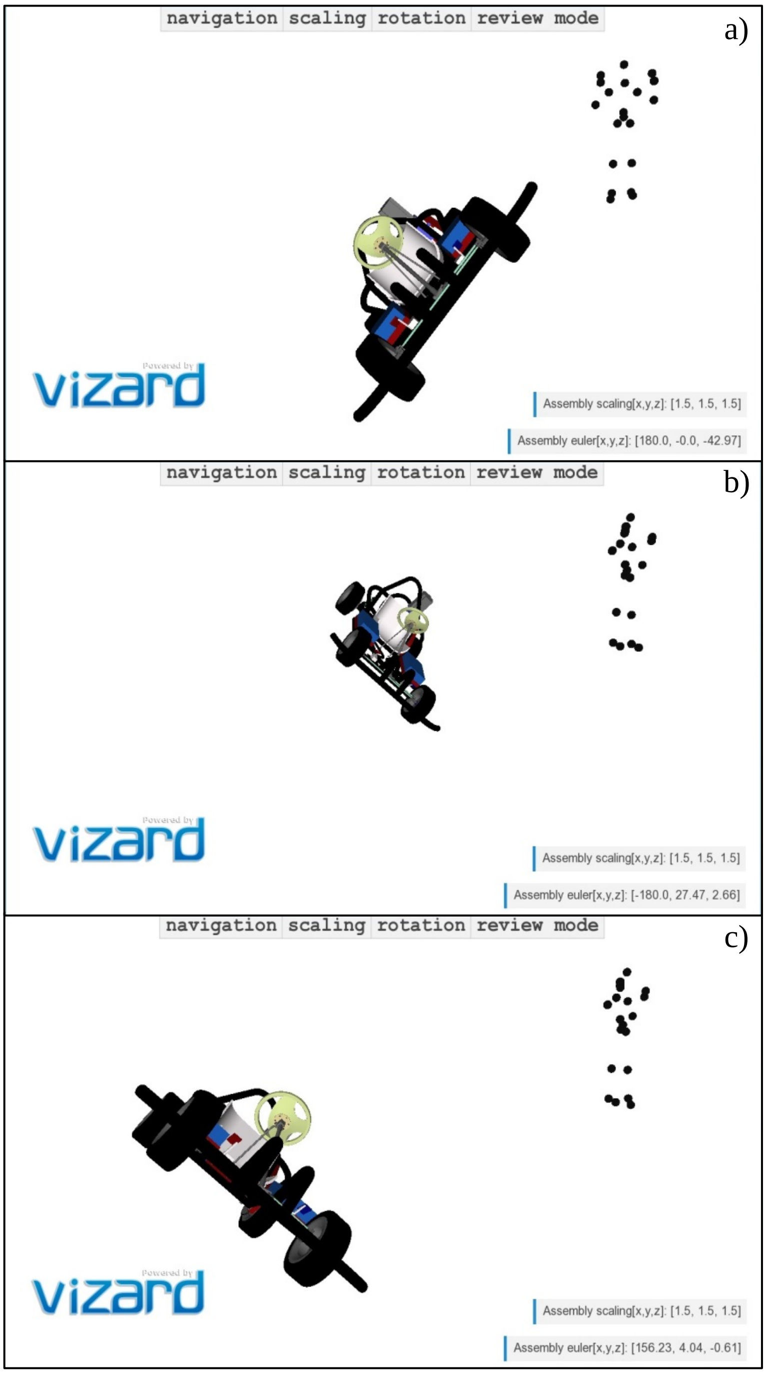


Figure 5-9 Assembly review rotation of kart

Screenshots of the detail design review process are shown in Figures 5-10, 5-11 and 5-12. Figure 5-10 exhibits the rotation operation in the detail review mode. The model can be rotated from x, y, z axes applying rotation hand gestures. Figure 5-10a shows the combination of x and y axes rotations of the frame module. Figure 5-10b is an x axis rotation and Figure 5-10c is a rotation of the shaft model in the frame module.

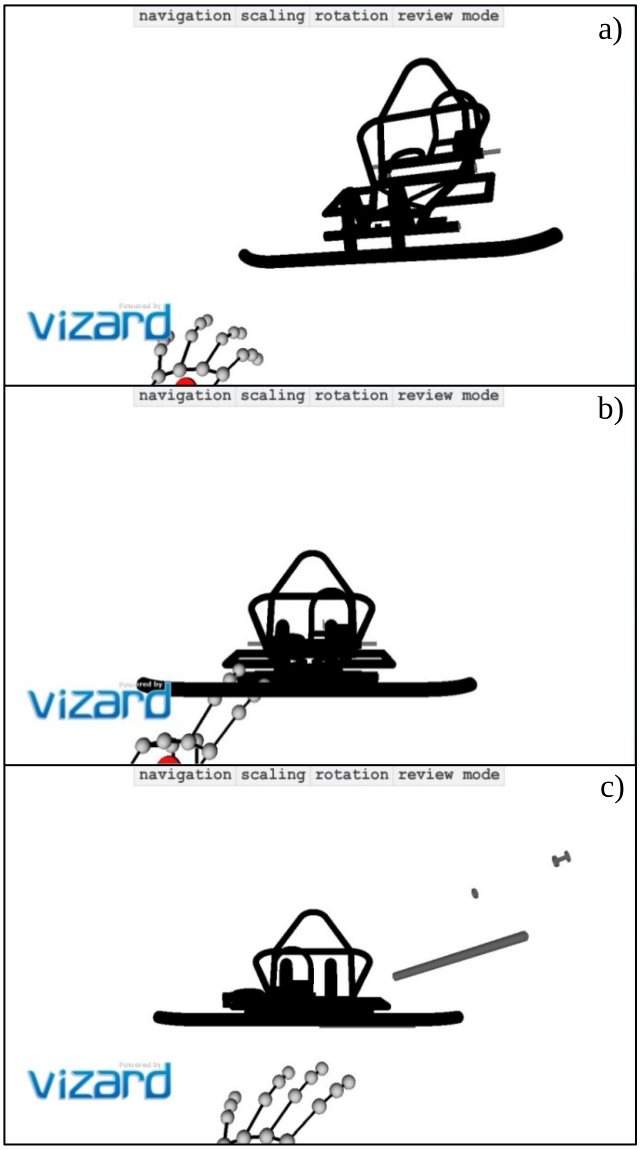


Figure 5-10 Detail review rotations of kart

Figure 5-11 shows scaling operations of the detail review to scale up and down the detail model to have a close look of the design.

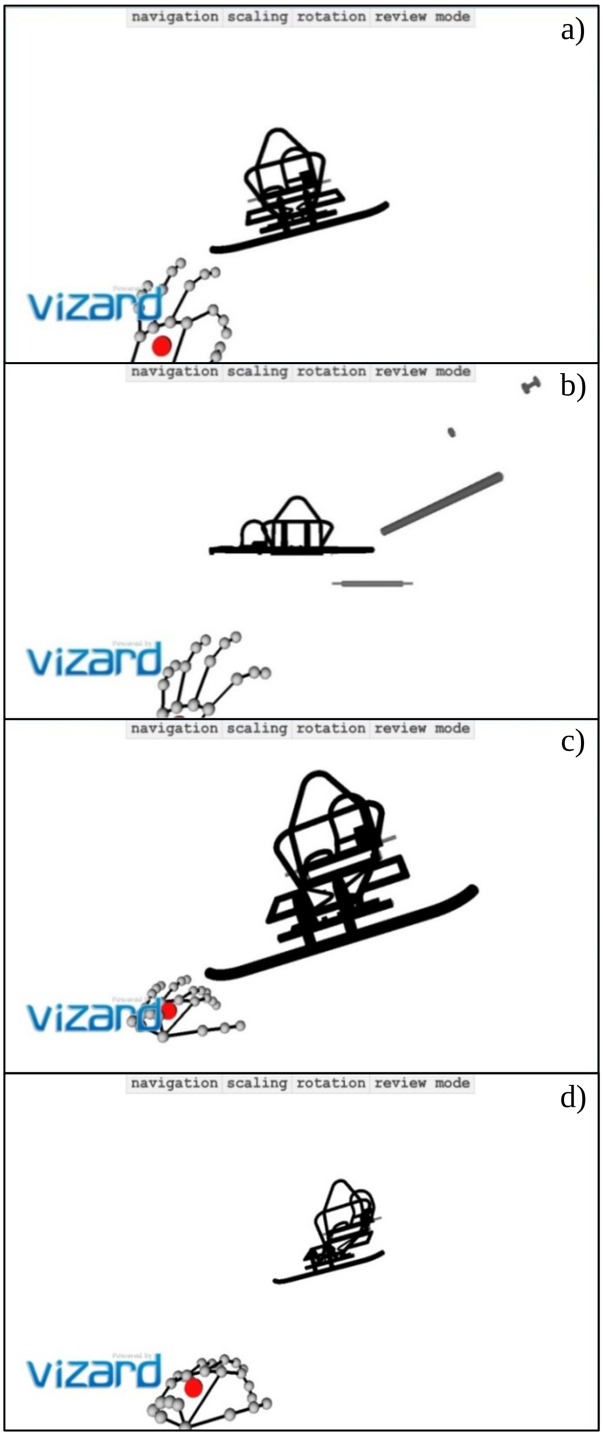


Figure 5-11 Detail review scaling of kart

Figure 5-12 shows translation operations for the detail review. The selected parts can be translated to different positions. Figure 5-12b presents the translation scene to grab the front frame. When the detail part is grabbed, the color will turn to red. When the grabbed part is released, the red color will be cleared. The grabbed part will turn into green when it moves to the original position. User can translate different parts in the module for a detail review.

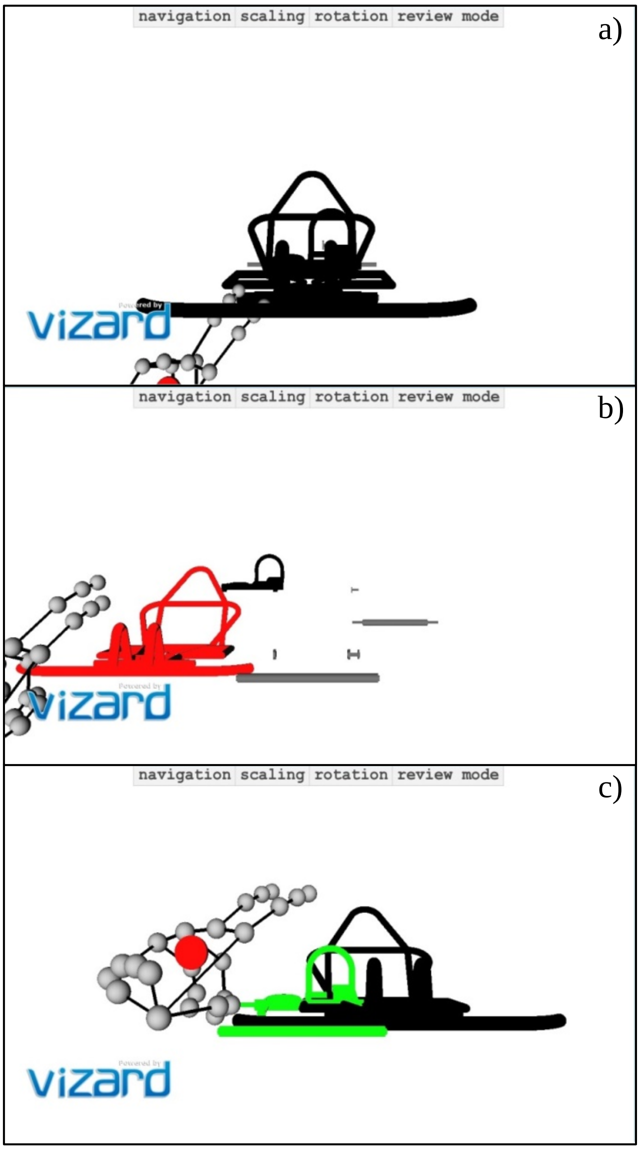


Figure 5-12 Detail review translation of kart

5.4.2. Gearbox model design review

The assembly review of the gearbox model is shown in Figures 5-13, 5-14 and 5-15. Figure 5-13 shows the navigation operation of the assembly. Figure 5-13a presents the gearbox model from the further view. Figure 5-13b shows the gearbox from the right view and Figure 5-13c displays the model from a close view.

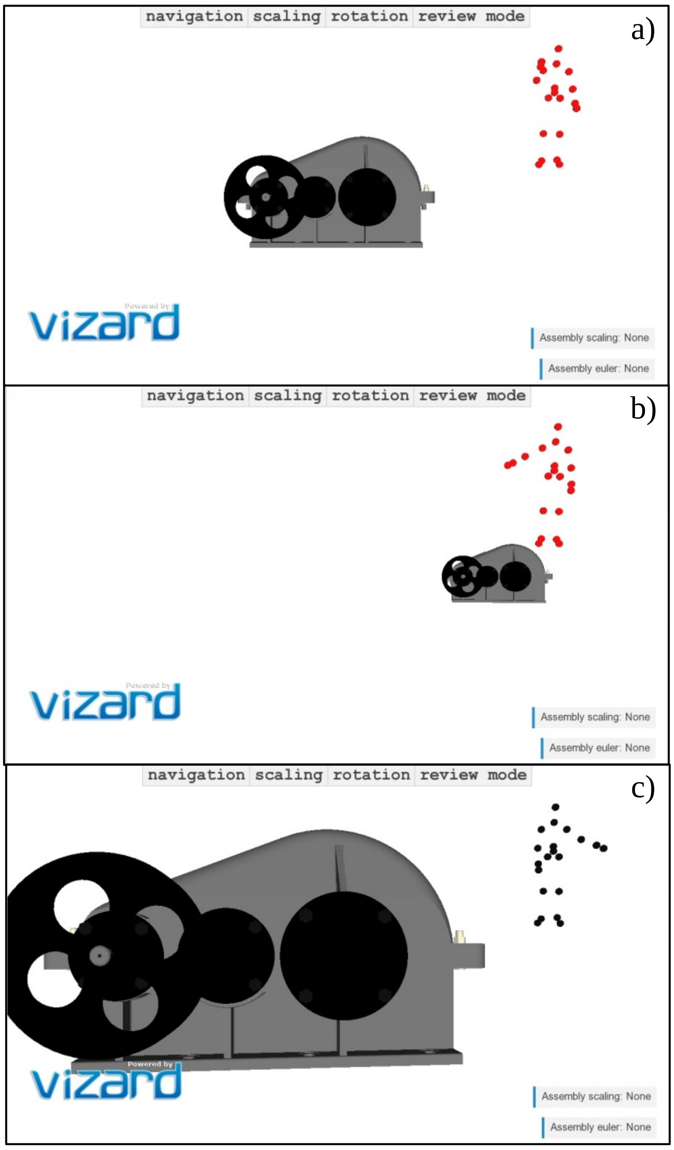


Figure 5-13 Assembly review navigation of gearbox

Figure 5-14 presents the scaling operation to scale up and down the assembly product model. Figures 5-14a and 5-14b present the assembly model after performing scaling up and down operations, respectively. Figure 5-8c shows the scaling up operation from a different viewpoint.

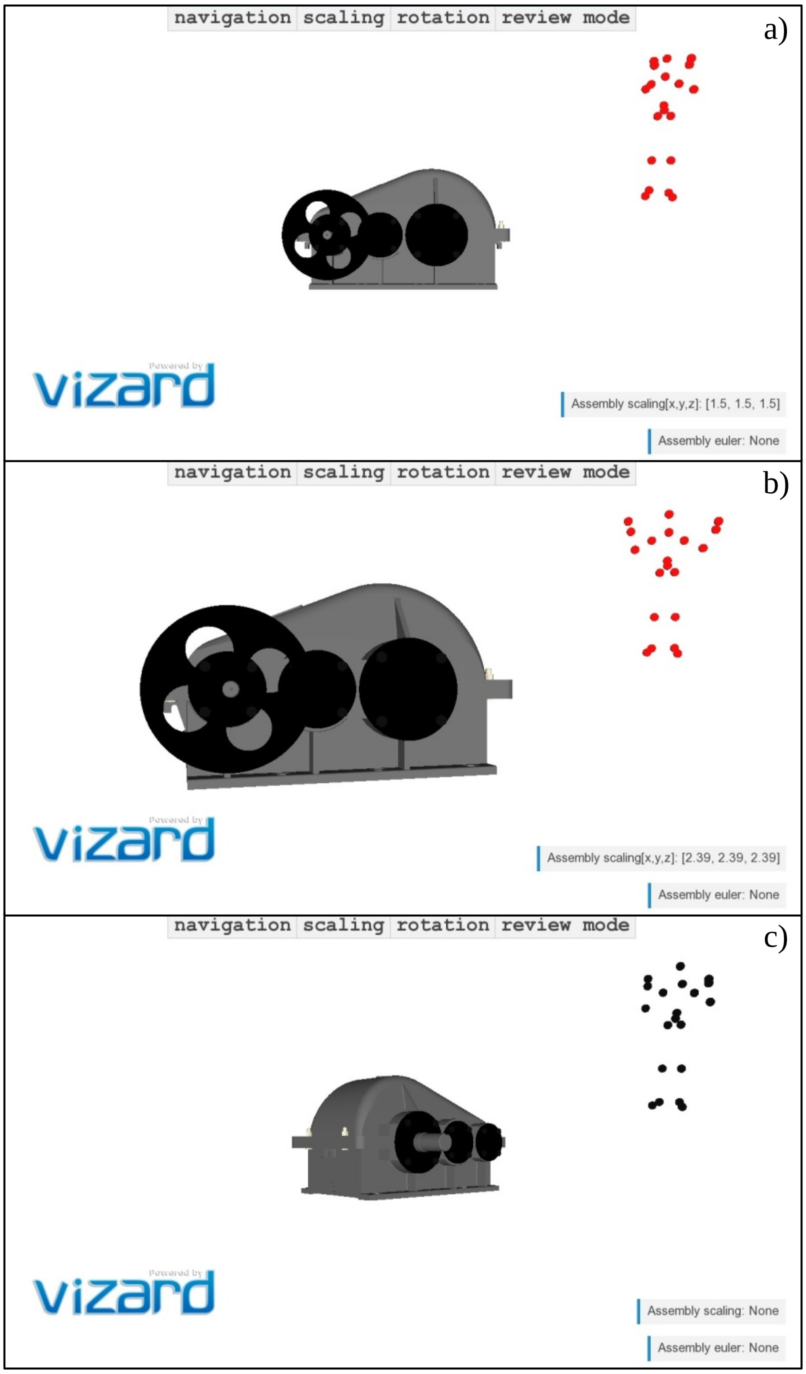


Figure 5-14 Assembly review scaling of gearbox

Figure 5-15 exhibits the rotation operation for the model to be rotated from x, y, z axes applying the rotation gesture. Figures 5-15a and 5-15b show the z axis rotation from positive and negative directions. Figure 5-15c is a combination of x, y and z axes rotations.

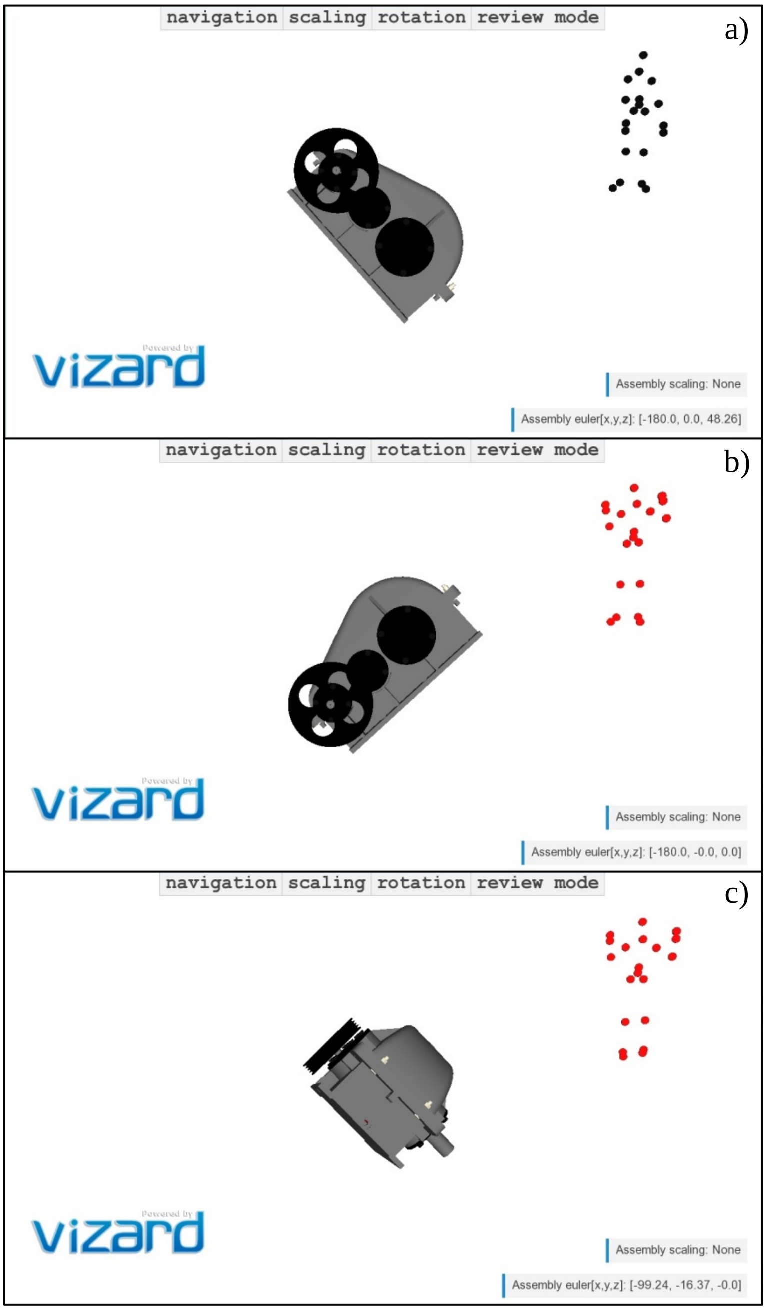


Figure 5-15 Assembly review rotation of gearbox

Figures 5-16, 5-17 and 5-18 display the detail review of the output shaft module of the gearbox model. Figure 5-16 shows the rotation operation of the output shaft module. The model can be rotated from x, y, z axes applying rotation hand gestures. Figure 5-16b is an x axis rotation and Figure 5-16c is a rotation scene of partly assembly of the output shaft module.

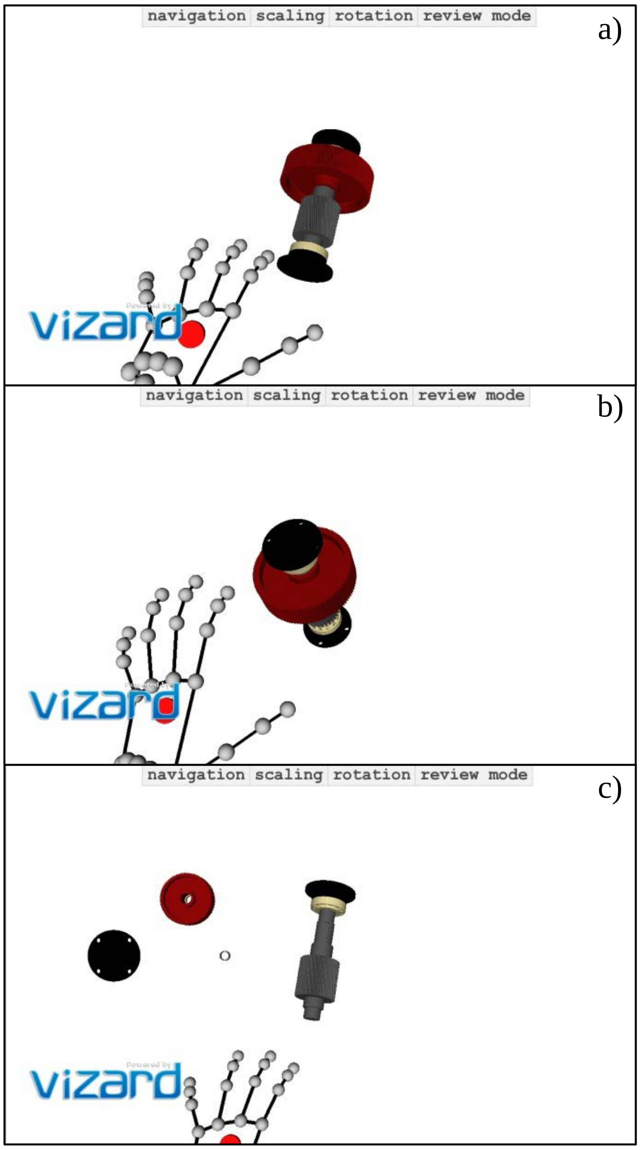


Figure 5-16 Detail review rotations of gearbox

Figure 5-17 shows scaling operations to scale up and down the detail model to have a close look of the design. Figures 5-17a and 5-17b scale up and down the output shaft assembly. Figures 5-17c and 5-17d display the detail parts.

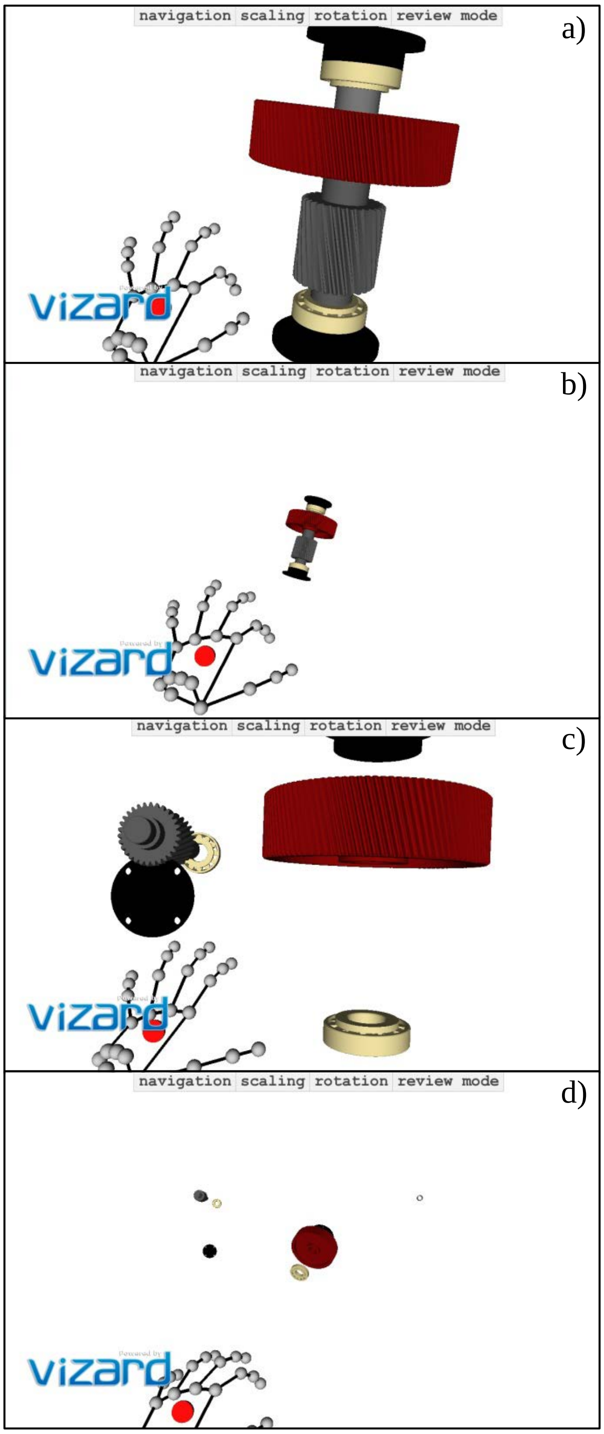


Figure 5-17 Detail review scaling of gearbox

Figure 5-18 shows translation operations. The selected parts can be translated to different positions. Figure 5-18b presents the translation scene to grab the flange. When the flange is grabbed, the color will turn to red. When the grabbed part is released, the red color will be cleared. The grabbed part will turn into green when it moves to the original position. User can translate different parts in the module for a detail review.

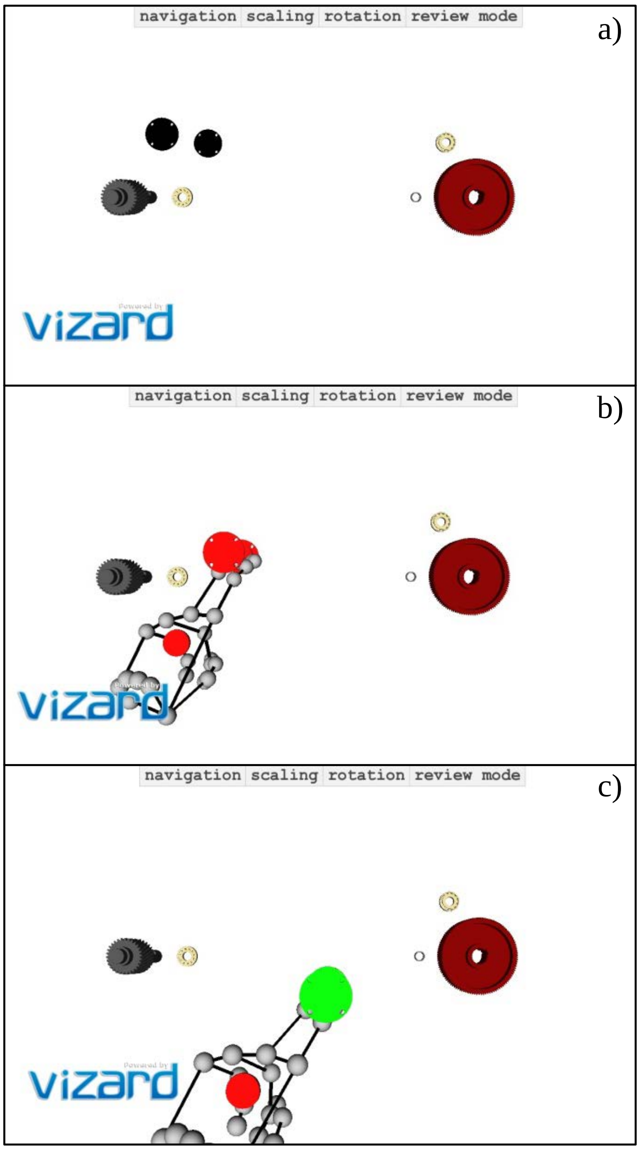


Figure 5-18 Detail review translation of gearbox

5.5. Comparison of design review systems

The evaluation of the design review system can find if the system fulfills the requirement of user experience. The analysis of the user experience result is helpful for the system improvement. Schrepp et al. proposed a User Experience Questionnaire (UEQ) method to compare the user experience of products (Schrepp, Hinderks and Thomaschewski, 2014). There are many research activities using the UEQ to evaluate products (Dicke *et al.*, 2012; Santoso *et al.*, 2014, 2017; Nawaz *et al.*, 2015). For the evaluation of the proposed design review system, a comparison study among the body gesture-based design review, hand gesture-based design review and body and hand gesture-based design review is made using the UEQ method (Hinderks, Schrepp and Thomaschewski, 2016)..

5.5.1. User test

Although the design review systems have been developed using gestures for a more natural and intuitive way than the computer and mouse in reviewing product model, the evaluation of the effectiveness and usability for the gesture-based design review systems has not been conducted yet. As the effectiveness and usability will affect users' understanding while reviewing the product model, the purposes of this user test are: (1) to understand users' experience in using gestures in a design review, and (2) to provide suggestions for improvement of the gesture-based design review system.

Twelve users participated in the user test were asked to complete the user experience questionnaire. All the participants had used the CAD software before and were familiar with CAD commands. Before the user test, a training process of the three systems was introduced.

During the training process, they were taught to perform gestures to review the models and they can ask question about the gesture operations in order to learn and remember gestures. The questionnaire contains two quality categories of the user experience including pragmatic quality (goal-directed) and hedonic quality (not goal-directed) aspects as shown in Figure 5-19. The questionnaire consists of 26 items categorized into six scales: attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty. Perspicuity, efficiency and dependability are pragmatic quality aspects. Stimulation and novelty are hedonic quality aspects. Attractiveness is a pure valence dimension. The explanation of the six scales is in Table 5-3. Reliability of the UEQ scales is typically high and two sample t-tests will be done to calculate the statistical significant difference between two comparing systems. Sample items of the questionnaire can be found in Table 5-4. The statistical results of the comparison between body gesture-based and body and hand gesture-based design review systems are shown in Table 5-5 and Figure 5-20. The statistical results of the comparison between the hand gesture-based and body and hand gesture-based design review systems are shown in Table 5-6 and Figure 5-21.

Table 5-3 Scale explanation (Santoso *et al.*, 2017)

Scale	Definition
Attractiveness	The overall impression of the system.
Perspicuity	The easiness to get familiar with the system and learn how to use the system.
Efficiency	The users can review the design model thoroughly without any unnecessary effort.
Dependability	The users feel in control of the human computer interaction.
Stimulation	The users feel exciting and motivating to use the system.
Novelty	The system is innovative and creative and it can catch users' interest.

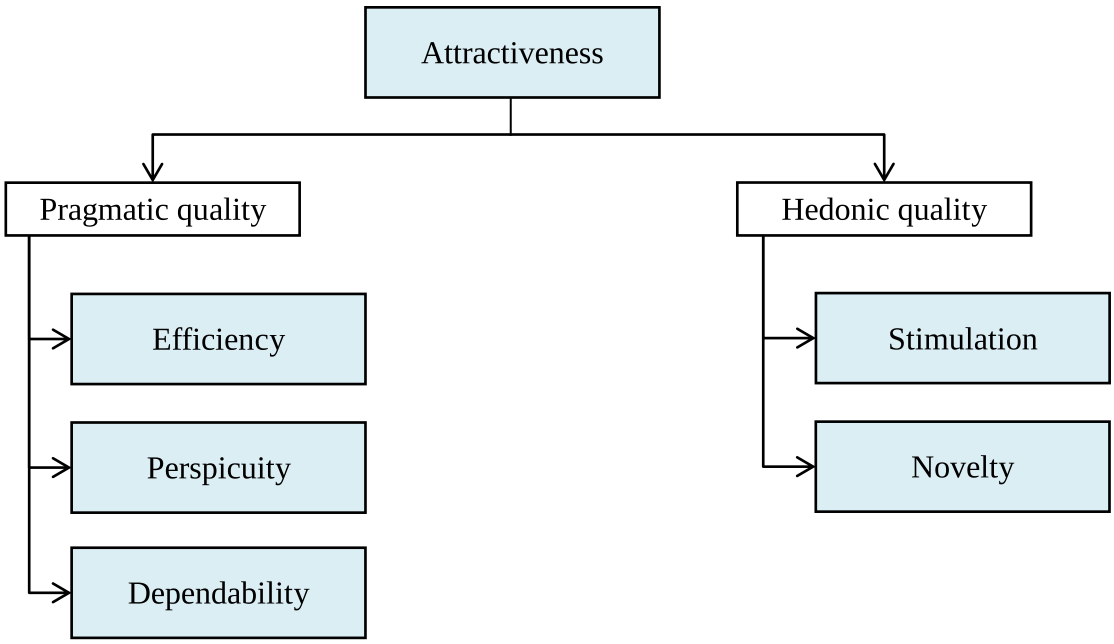


Figure 5-19 Scale classification (Santoso *et al.*, 2017)

Table 5-4 Sample item of questionnaire

Attractiveness		
Attractive		Unattractive
Bad		Good
Unlikable		Pleasing
Unpleasant		Pleasant
Unattractive		Attractive
Unfriendly		Friendly

The items are scaled from 1 to 7. Thus, 1 represents the most negative answer, 4 a neutral answer, and 7 the most positive answer.

Table 5-5 Comparison of gesture-based design review system I

Scale	Data Set 1			Data Set 2			P-value
	Mean	STD	N	Mean	STD	N	
Attractiveness	1.43	0.22	12	1.90	0.22	12	0
Perspicuity	1.46	0.40	12	1.46	0.30	12	1.00
Efficiency	0.94	0.28	12	1.94	0.24	12	0
Dependability	1.77	0.27	12	1.88	0.42	12	0.48
Stimulation	1.90	0.27	12	1.85	0.29	12	0.72
Novelty	1.50	0.34	12	1.71	0.23	12	0.09

Data Set 1: Body gesture-based, Data Set 2: Body and hand gesture-based

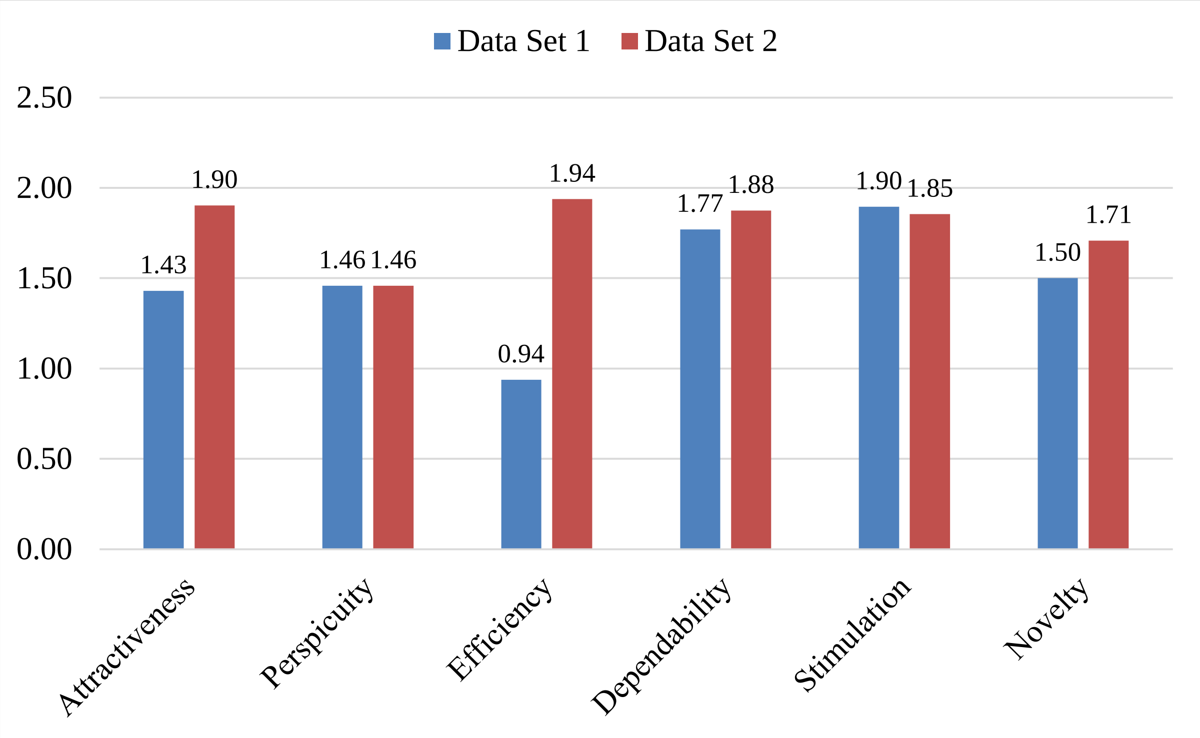


Figure 5-20 Comparison of gesture-based design review system I

Table 5-5 shows the number of the sample, mean value, standard deviation of six comparing scales. P-values of the attractiveness and efficiency are smaller than 0.05, which means that there are significant differences in the feedback of the scale means of the attractiveness and efficiency. Though there are slight differences between the means of the rest of the four scales in Figure 5-20, the p-values of perspicuity, dependability, stimulation and novelty are bigger than 0.05 that there are no significant differences. From the statistical results, we can come to a conclusion that users are more impressed by the body and hand gesture-based design review system because of the thorough reviewing function that users can achieve both assembly and detail review. Considering the easiness of learning and using of the system, motivation and innovation, the body gesture-based design review system is comparable to the body and hand gesture-based design review system.

Table 5-6 Comparison of gesture-based design review system II

Scale	Data Set 1			Data Set 2			P-value
	Mean	STD	N	Mean	STD	N	
Attractiveness	1. 65	0. 21	12	1. 90	0. 22	12	0. 0087
Perspicuity	1. 13	0. 31	12	1. 46	0. 30	12	0. 0137
Efficiency	0. 94	0. 37	12	1. 94	0. 24	12	0
Dependability	1. 38	0. 35	12	1. 88	0. 42	12	0. 0044
Stimulation	1. 65	0. 34	12	1. 85	0. 29	12	0. 1244
Novelty	1. 60	0. 31	12	1. 71	0. 23	12	0. 3640

Data Set 1: Hand gesture-based, Data Set 2: Body and hand gesture-based

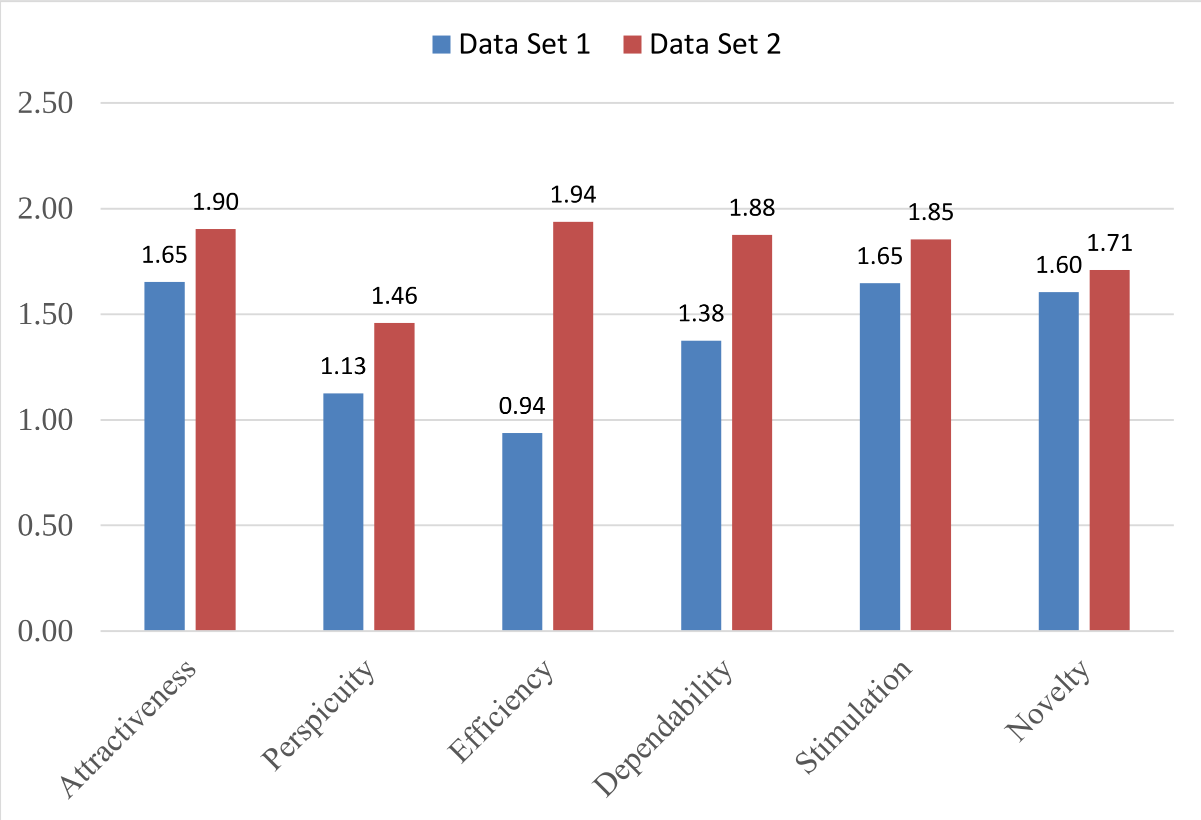


Figure 5-21 Comparison of gesture-based design review system II

Table 5-6 presents the number of the sample, mean value, standard deviation of six comparing scales. The p-values of attractiveness, perspicuity, efficiency and dependability are smaller than 0.05, which means that there are significant differences in the feedback of the scale means of attractiveness, perspicuity, efficiency and dependability. The p-values of stimulation and novelty are bigger than 0.05 that there are no significant differences. From the statistical results, we can come to two valuable conclusions that users are more impressed by the body and hand gesture-based design review system because of the thorough reviewing function and body gestures can provide user with more feeling of immersion and it can make users feel in control of the human computer interaction. Considering the excitement, innovation and creativity of the system, users are interested in using the gesture-based design review systems.

Chapter 6

Conclusions and future work

6.1. Research summary

Based on the literature review of human computer interaction (HCI) devices, virtual reality (VR) technology and gesture recognition methods, advantages and disadvantages of different HCI devices for the gesture recognition are compared and analyzed. The Microsoft Kinect and Leap Motion Controller (LMC) are chosen for the gesture recognition using a template-based method. Vizard VR software is selected as the platform to provide users immersive feeling during the design review process. Through the analysis of applications in the gesture recognition, combining gestures with proper applications in CAD/CAE areas is still under development. As Microsoft Kinect comes with specific body gesture recognition features and LMC is targeted for the hand gesture recognition, body gesture-based and hand gesture-based design review systems are developed separately to compare the design review functions and system performance. Six static gestures and six dynamic gestures are defined to represent movements in the navigation, translation, rotation, scaling, disassembly and assembly commands in the body gesture-based design review system. Recognition rates of the designed body gestures are over 80%. Users can review the model from the product assembly to modules. Four dynamic gestures representing translation, rotation, scaling and switching commands are designed for the hand gesture-based design review system with the average gesture recognition rates of over 80%. Users can review the model from product modules to detail parts. User tests are conducted for both systems. The result of the user test indicates that gestures are more natural and intuitive comparing with the traditional computer mouse and keyboard as input. Through the analysis of the two design review systems, a body and hand gesture-based design review system is proposed to combine the design review features of both systems to achieve a complete design review process from product assembly to detail parts. An UEQ survey is conducted to evaluate the new system. The result indicates that the new system can provide users complete design review process. The resulting project provides an immersive, user-friendly platform for the CAD/CAE design review process through the integration of VR and gesture recognition technology. The natural and intuitive interface helps users review product design in a quick, efficient and intuitive way.

6.2. Summary of contributions

The contributions of this research are as follows:

(1) Six static and six dynamic body gestures are designed to map CAD commands with body gestures for the assembly design review.

(2) Virtual computer mouse control is proposed to operate the menu in order to eliminate the gesture interference in the body gesture-based design review system. Four of the data smoothing methods are compared including Gaussian filter, average, weighted average#1, and weighted average#2 to smooth data transmitted from the Kinect sensor and to improve the menu selection accuracy.

(3) The Vizard VR environment is integrated with the design review system to provide users an immersive, intuitive and natural interface.

(4) Four dynamic hand gestures are designed to map CAD commands with hand gestures for the detail design review.

(5) A template-based mapping method is applied for both body and hand gesture recognitions with a recognition rate of over 80%.

(6) A design review system combining body and hand gestures is presented to achieve both assembly and detail design reviews.

6.3. Future work

In the current research, all of the gestures are predefined and a training process is mandatory for users before using the design review system. A machine learning algorithm will be considered to expand the system with customized gestures. Also, the CAD designing process in the VR environment will be considered to reduce efforts to convert data format from CAD to VR systems.

In the body and hand gesture-based design review system, the Kinect sensor and LMC are placed in different locations and users need to move their positions for the detail review. Head Mounted Device (HMD) will be considered for a full immersive visualization in the virtual environment. LMC can be placed on the forehead of HMD to eliminate users' movements.

Papers published related to this research

Yu Xiao, Qingjin Peng, Body gesture-based user interaction in design review, Proceedings of the ASME 2017 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, IDETC/CIE 2017, August 06-09, 2017, Cleveland, Ohio, USA, DETC2017-67415.

Yu Xiao, Qingjin Peng, A hand gesture-based interface for design review using Leap Motion Controller, Proceedings of the ICED 2017 International Conferences on Engineering Design, ICED 2017, August 21-25, 2017, Vancouver, British Columbia, Canada.

References

Artyukhin, S. G. and Mestetskiy, L. M. (2015) ‘Dactyl Alphabet Gesture Recognition in a Video Sequence Using Microsoft Kinect’, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-5/W6(May), pp. 83–86. doi: 10.5194/isprsarchives-XL-5-W6-83-2015.

Avola, D., Petracca, A., Placidi, G., Spezialetti, M., Cinque, L. and Levialdi, S. (2014) ‘Markerless hand gesture interface based on LEAP motion controller’, *20th International Conference on Distributed Multimedia Systems, DMS 2014*, pp. 260–266. Available at: http://www.scopus.com/inward/record.url?eid=2-s2.0-84923923300&partnerID=40&md5=a366697ac875af033e16a7be4e6ab77a.

Babu, S. S. S., Jaiswal, P., Esfahani, E. T. and Rai, R. (2014) ‘Sketching in Air: A Single Stroke Classification Framework’, in *ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. ASME. doi: 10.1115/DETC2014-34065.

Barfield, W., Hendrix, C. and Bystrom, K. (1997) ‘Visualizing the Structure of Virtual Objects Using Head Tracked Stereoscopic Displays’, *Proceedings of IEEE 1997 Annual International Symposium on Virtual Reality*, pp. 114–120. doi: 10.1109/VRAIS.1997.583052.

Berta, J. (1999) ‘Integrating VR and CAD’, *IEEE Computer Graphics and Applications*, 19(5), pp. 14–19. doi: 10.1109/38.788793.

Billinghurst, M. (1998) ‘Put that where? voice and gesture at the graphics interface’, *ACM SIGGRAPH Computer Graphics*, 32(4), pp. 60–63. doi: http://doi.acm.org/10.1145/307710.307730.

Buchmann, V., Violich, S., Billinghurst, M. and Cockburn, A. (2004) ‘FingARtips: gesture based direct manipulation in Augmented Reality’, in *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pp. 212–221. doi: 10.1145/988834.988871.

Buisine, S., Besacier, G., Aoussat, A. and Vernier, F. (2012) ‘How do interactive tabletop systems influence collaboration?’, *Computers in Human Behavior*, 28(1), pp. 49–59. doi: 10.1016/j.chb.2011.08.010.

Camgoz, N. C., Kindiroglu, A. A. and Akarun, L. (2015) ‘Gesture Recognition Using Template Based Random Forest Classifier’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8927, pp. 570–583. doi: 10.1007/978-3-319-16199-0.

Carrera, K. C. P., Erise, A. P. R., Abrena, E. M. V, Colot, S. J. S. and Tolentino, R. E. (2014) ‘Application of Template Matching Algorithm for Dynamic Gesture Recognition of American Sign Language Finger Spelling and Hand Gesture’, *Asia Pacific Journal of Multidisciplinary Research*, 2(4), pp. 154–158.

Cassola, F., Morgado, L., de Carvalho, F., Paredes, H. and Fonseca, B. (2014) ‘Online-Gym: A 3D Virtual Gymnasium Using Kinect Interaction’, in *Procedia Technology*. Elsevier B.V., pp. 130–138. doi: 10.1016/j.protcy.2014.02.017.

Chang, C.-Y., Lange, B., Zhang, M., Koenig, S., Requejo, P., Somboon, N., Sawchuk, A. and

Rizzo, A. (2012) 'Towards Pervasive Physical Rehabilitation Using Microsoft Kinect', in *Proceedings of the 6th International Conference on Pervasive Computing Technologies for Healthcare*. doi: 10.4108/icst.pervasivehealth.2012.248714.

Dave, D., Chowriappa, A. and Kesavadas, T. (2013) 'Gesture interface for 3D CAD modeling using Kinect', *Computer-Aided Design and Applications*, 10(4), pp. 663–669. doi: 10.3722/cadaps.2013.663-669.

Dicke, C., Jakus, G., Tomažič, S. and Sodnik, J. (2012) 'On the Evaluation of Auditory and Head-up Displays While Driving', in *ACHI 2012 : The Fifth International Conference on Advances in Computer-Human Interactions On*, pp. 200–203.

Ding, I. J. and Chang, C. W. (2016) 'An adaptive hidden Markov model-based gesture recognition approach using Kinect to simplify large-scale video data processing for humanoid robot imitation', *Multimedia Tools and Applications*, 75(23), pp. 15537–15551. doi: 10.1007/s11042-015-2505-9.

EI-Mounayri, H., Rogers, C. and Fernandez, E. (2016) 'Assessment of STEM e-Learning in an Immersive Virtual Reality (VR) Environment', in *ASEE's 123rd Annual Conference&Exposition*. Available at: https://www.asee.org/file_server/papers/attachment/file/0006/9319/IVR_ASEE_2016_V3.pdf

Fechter, M., Miehling, J., Stangl, T. and Wartzack, S. (2014) 'Intuitive Virtual Reality- CAD Assembly System', *Proceedings of the 13th International Design Conference DESIGN 2014*, pp. 1053–1062.

Fels, S. S., Pritchard, B. and Lenters, A. (2009) 'ForTouch: A Wearable Digital Ventriloquized Actor', in *Proceedings of the International Conference on New Interfaces for*

Musical Expression, pp. 274–275. Available at: http://www.nime.org/proceedings/2009/nime2009_274.pdf.

Fillatreau, P., Fourquet, J. Y., Le Bolloc'H, R., Cailhol, S., Datas, A. and Puel, B. (2013) 'Using virtual reality and 3D industrial numerical models for immersive interactive checklists', *Computers in Industry*. Elsevier B.V., 64(9), pp. 1253–1262. doi: 10.1016/j.compind.2013.03.018.

Fiorentino, M., Radkowski, R., Stritzke, C., Uva, A. E. and Monno, G. (2012) 'Design review of CAD assemblies using bimanual natural interface', *International Journal on Interactive Design and Manufacturing (IJIDeM)*, 7(4), pp. 249–260. doi: 10.1007/s12008-012-0179-3.

Francese, R., Passero, I. and Tortora, G. (2012) 'Wiimote and Kinect: Gestural User Interfaces add a Natural third dimension to HCI', in *Proceedings of the International Working Conference on Advanced Visual Interfaces - AVI '12*, p. 116. doi: 10.1145/2254556.2254580.

Freeman, I. J. and Coburn, J. Q. (2016) 'Cad Integration in Virtual Reality Design Reviews for Improved', in *Proceedings of the ASME 2016 International Mechanical Engineering Congress and Exposition*, pp. 1–10.

Gallo, L., Placitelli, A. P. and Ciampi, M. (2011) 'Controller-free exploration of medical image data: Experiencing the Kinect', in *2011 24th International Symposium on Computer-Based Medical Systems (CBMS)*. doi: 10.1109/CBMS.2011.5999138.

Grandhi, S. A., Joue, G. and Mittelberg, I. (2011) 'Understanding naturalness and intuitiveness in gesture production', in *Proceedings of the 2011 annual conference on Human factors in computing systems*, p. 821. doi: 10.1145/1978942.1979061.

Guerra, A. L., Gidel, T. and Vezzetti, E. (2016) 'Toward a Common Procedure Using Likert and Likert-Type Scales in Small Groups Comparative Design Observations', in *Proceedings of the 2016 14th International Design Conference*, pp. 23–32.

Guna, J., Jakus, G., Pogacnik, M., Tomazic, S. and Sodnik, J. (2014) 'An analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking', *Sensors (Switzerland)*, 14(2), pp. 3702–3720. doi: 10.3390/s140203702.

Hinderks, A., Schrepp, M. and Thomaschewski, J. (2016) *UEQ-User Experience Questionnaire*. Available at: http://www.ueq-online.org/.

Hsieh, W., Chen, C., Wang, S., Tan, S., Hwang, Y. and Chen, S. (2014) 'Virtual reality system based on Kinect for the elderly in fall prevention', *Technology and Health Care*, 22(134), pp. 27–36. doi: 10.3233/THC-130769.

Ishii, K., Soga, M. and Taki, H. (2011) 'Proposal and development of reversal motion skill leaning support environment', in *2011 International Conference on Information Technology Based Higher Education and Training, ITHET 2011*. doi: 10.1109/ITHET.2011.6018690.

Kadous, M. W. (2002) *Temporal classification: Extending the classification paradigm to multivariate time series*. Available at: http://www.cse.unsw.edu.au/~waleed/phd/phd.pdf.

Kerefeyn, S. and Maleshkov, S. (2015) 'Manipulation of virtual objects through a LeapMotion optical sensor', *International Journal of Computer Science Issues*, 12(5), pp. 52–57.

Keskin, C., Erkan, A. and Akarun, L. (2003) 'Real time hand tracking and 3D gesture recognition for interactive interfaces using HMM', *Icann/Iconipp*, pp. 26–29.

Kessler, G. D., Hodges, L. F. and Walker, N. (1995) 'Evaluation of the CyberGlove as a whole-hand input device', *ACM Transactions on Computer-Human Interaction*, 2(4), pp. 263–283. doi: 10.1145/212430.212431.

Kim, D., Hilliges, O., Izadi, S., Butler, A. D., Chen, J., Oikonomidis, I. and Olivier, P. (2012) 'Digits: Freehand 3D Interactions Anywhere Using a Wrist-Worn Gloveless Sensor', in *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pp. 167–176. doi: 10.1145/2380116.2380139.

Kim, S. K., Hong, S. and Kim, D. (2009) 'A walking motion imitation framework of a humanoid robot by human walking recognition from IMU motion data', in *9th IEEE-RAS International Conference on Humanoid Robots, HUMANOIDS09*, pp. 343–348. doi: 10.1109/ICHR.2009.5379552.

Kinect Robotic (2012). Available at: http://blog.csdn.net/mdl13412/article/details/8025680 (Accessed: 17 January 2016).

Kováč, J., Ďurovský, F. and Varga, J. (2014) 'Integrated System of Mixed Virtual Reality Based on Data Glove CyberGlove II and Robotic Arm MechaTE Robot', *Applied Mechanics and Materials*, 611, pp. 239–244. doi: 10.4028/www.scientific.net/AMM.611.239.

Kumar, P., Verma, J. and Prasad, S. (2012) 'Hand Data Glove: A Wearable Real-Time Device for Human-Computer Interaction', *International Journal of Advanced Science and Technology*, 43, pp. 15–26. doi: 10.1109/RAIT.2012.6194548.

Le, V. B., Nguyen, A. T. and Zhu, Y. (2014) 'Hand Detecting and Positioning Based on Depth Image of Kinect Sensor', *International Journal of Information and Electronics Engineering*, 4(3), pp. 176–179. doi: 10.7763/IJIEE.2014.V4.430.

Leap Motion (2015). Available at: https://www.leapmotion.com/ (Accessed: 27 July 2017).

Lee, S.-S., Chae, J., Kim, H., Lim, Y. and Lee, K. (2013) ‘Towards more natural digital content manipulation via user freehand gestural interaction in a living room’, in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing - UbiComp ’13*, p. 617. doi: 10.1145/2493432.2493480.

Mahbub, U., Imtiaz, H., Roy, T., Rahman, M. S. and Ahad, M. A. R. (2013) ‘A template matching approach of one-shot-learning gesture recognition’, *Pattern Recognition Letters*. Elsevier B.V., 34(15), pp. 1780–1788. doi: 10.1016/j.patrec.2012.09.014.

Marin, G., Dominio, F. and Zanuttigh, P. (2014) ‘Hand gesture recognition with leap motion and kinect devices’, in *2014 IEEE International Conference on Image Processing, ICIP 2014*, pp. 1565–1569. doi: 10.1109/ICIP.2014.7025313.

Mccartney, R., Yuan, J. and Bischof, H. (2015) ‘Gesture Recognition with the Leap Motion Controller’, in *International Conference on Image Processing, Computer Vision, and Pattern Recognition*, pp. 3–9.

Miranda, L., Vieira, T., Martinez, D., Lewiner, T., Vieira, A. W. and Campos, M. F. M. (2012) ‘Real-time gesture recognition from depth data through key poses learning and decision forests’, *Brazilian Symposium of Computer Graphic and Image Processing*, pp. 268–275. doi: 10.1109/SIBGRAPI.2012.44.

Mistry, P. and Maes, P. (2009) ‘SixthSense: A Wearable Gestural Interface’, in *ACM SIGGRAPH ASIA 2009 Sketches*, p. 11:1. doi: 10.1145/1667146.1667160.

Mohandes, M. A. (2013) ‘Recognition of Two-Handed Arabic Signs Using the CyberGlove’, *Arabian Journal for Science and Engineering*, 38(3), pp. 669–677. doi: 10.1007/s13369-012-0378-z.

Mujber, T. S., Szecsi, T. and Hashmi, M. S. J. (2004) ‘Virtual reality applications in manufacturing process simulation’, *Journal of Materials Processing Technology*, 155–156(1–3), pp. 1834–1838. doi: 10.1016/j.jmatprotec.2004.04.401.

Nagarajan, S. and Subashini, T. S. (2015) ‘Image Based Hand Gesture Recognition using Statistical Features and Soft Computing Techniques’, *International Journal of Emerging Technology and Advanced Engineering*, 5(7), pp. 476–482.

Nanjundaswamy, V. G., Kulkarni, A., Chen, Z., Jaiswal, P., Shankar, S. S. and Verma, A. (2013) ‘Intuitive 3D Computer-aided design (CAD) System with Multimodal Interfaces’, in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. 1–11.

Nawaz, A., Helbostad, J. L., Chiari, L., Chesani, F. and Cattelani, L. (2015) ‘User Experience (UX) of the Fall Risk Assessment Tool (FRAT-up)’, in *2015 IEEE 28th International Symposium on Computer-Based Medical Systems*, pp. 19–22. doi: 10.1109/CBMS.2015.63.

Nguyen-Dinh, L. Van, Roggen, D., Calatroni, A. and Tröster, G. (2012) ‘Improving online gesture recognition with template matching methods in accelerometer data’, in *International Conference on Intelligent Systems Design and Applications, ISDA*, pp. 831–836. doi: 10.1109/ISDA.2012.6416645.

Palacios, J. M., Sagues, C., Montijano, E. and Llorente, S. (2013) ‘Human-computer interaction based on hand gestures using RGB-D sensors’, *Sensors (Basel, Switzerland)*, 13(9), pp. 11842–11860. doi: 10.3390/s130911842.

Pavlovic, V. I., Sharma, R. and Huang, T. S. (1997) ‘Visual interpretation of hand gestures for human-computer interaction: a review’, *IEEE Transactions on Pattern Analysis and*

Machine Intelligence, 19(7), pp. 677–695. doi: 10.1109/34.598226.

Pontonnier, C., Dumont, G., Samani, A., Madeleine, P. and Badawi, M. (2014) ‘Designing and evaluating a workstation in real and virtual environment: Toward virtual reality based ergonomic design sessions’, *Journal on Multimodal User Interfaces*, 8(2), pp. 199–208. doi: 10.1007/s12193-013-0138-8.

Potter, L. E., Araullo, J. and Carter, L. (2013) ‘The Leap Motion controller’, in *Proceedings of the 25th Australian Computer-Human Interaction Conference on Augmentation, Application, Innovation, Collaboration - OzCHI ’13*, pp. 175–178. doi: 10.1145/2541016.2541072.

Pradipa, R. and Kavitha, M. S. (2014) ‘Hand Gesture Recognition – Analysis of Various Techniques , Methods and Their Algorithms’, *International Journal of Innovative Research in Science, Engineering and Technology*, 3(3), pp. 2003–2010.

Ren, Z., Yuan, J., Meng, J. and Zhang, Z. (2013) ‘Robust Part-Based Hand Gesture Recognition Using Kinect Sensor’, *IEEE Transactions on Multimedia*, 15(5), pp. 1110–1120. doi: 10.1109/TMM.2013.2246148.

Sabir, K. and Tabor, B. (2013) ‘The Molecular Control Toolkit : Controlling 3D Molecular Graphics via Gesture and Voice Se an’, in *2013 IEEE Symposium on Biological Data Visualization (BioVis)*, pp. 49–56.

Santoso, H. B., Isal, R. Y. K., Basaruddin, T., Sadita, L. and Schrepp, M. (2014) ‘Research-in-progress: User Experience Evaluation of Student Centered E-Learning Environment for Computer Science Program’, in *2014 3rd International Conference on User Science and Engineering (i-USEr)*, pp. 52–55. doi: 10.1109/IUSER.2014.7002676.

Santoso, H. B., Nurrohmah, I., Fadhilah, S. and Goodridge, W. H. (2017) ‘Evaluating and Redesigning the Self-Monitoring Tool’, *International Journal on Advanced Science, Engineering and Information Technology*, 7(1), p. 228. doi: 10.18517/ijaseit.7.1.1526.

Schrepp, M., Hinderks, A. and Thomaschewski, J. (2014) ‘Applying the User Experience Questionnaire (UEQ) in Different Evaluation Scenarios’, in Marcus, A. (ed.) *Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience: Third International Conference, DUXU 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part I*. Cham: Springer International Publishing, pp. 383–392. doi: 10.1007/978-3-319-07668-3_37.

Schuemie, M. J., van der Straaten, P., Krijn, M. and van der Mast, C. A. P. G. (2001) ‘Research on Presence in Virtual Reality: A Survey’, *CyberPsychology & Behavior*, 4(2), pp. 183–201. doi: 10.1089/109493101300117884.

Shin, M. C., Tsap, L. V. and Goldgof, D. B. (2004) ‘Gesture recognition using Bezier curves for visualization navigation from registered 3-D data’, *Pattern Recognition*, 37(5), pp. 1011–1024. doi: 10.1016/j.patcog.2003.11.007.

Shiratuddin, M. F. and Wong, K. W. (2011) ‘Non-contact multi-hand gestures interaction techniques for architectural design in a virtual environment’, in *2011 International Conference on Information Technology and Multimedia: ‘Ubiquitous ICT for Sustainable and Green Living’, ICIM 2011*. doi: 10.1109/ICIMU.2011.6122761.

Song, H., Chen, F., Peng, Q., Zhang, J. and Gu, P. (2017) ‘Improvement of user experience using virtual reality in open-architecture product design’, *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, p. 95440541771173.

doi: 10.1177/0954405417711736.

Song, J., Cho, S., Baek, S.-Y., Lee, K. and Bang, H. (2014) ‘GaFinC: Gaze and Finger Control interface for 3D model manipulation in CAD application’, *Computer-Aided Design*. Elsevier Ltd, 46, pp. 239–245. doi: 10.1016/j.cad.2013.08.039.

Stoerring, M., Moeslund, T., Liu, Y. and Granum, E. (2004) ‘Computer vision-based gesture recognition for an augmented reality interface’, in *International Conference on Visualization, Imaging and Image Processing*, pp. 766–771. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.73.104&rep=rep1&type=pdf.

Suma, E. a., Krum, D. M., Lange, B., Koenig, S., Rizzo, A. and Bolas, M. (2013) ‘Adapting user interfaces for gestural interaction with the flexible action and articulated skeleton toolkit’, *Computers and Graphics (Pergamon)*. Elsevier, 37(3), pp. 193–201. doi: 10.1016/j.cag.2012.11.004.

Sundar, H., Silver, D., Gagvani, N. and Dickinson, S. (2003) ‘Skeleton based shape matching and retrieval’, in *2003 Shape Modeling International*, pp. 130–139. doi: 10.1109/SMI.2003.1199609.

Tang, H. H., Lee, Y. Y. and Gero, J. S. (2011) ‘Comparing collaborative co-located and distributed design processes in digital and traditional sketching environments: A protocol study using the function-behaviour-structure coding scheme’, *Design Studies*. Elsevier Ltd, 32(1), pp. 1–29. doi: 10.1016/j.destud.2010.06.004.

Thakur, A. and Rai, R. (2015) ‘User Study of Hand Gestures for Gesture Based 3D Cad Modeling’, in *Proceedings of the ASME 2015 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, pp. 1–13. doi:

10.1115/DETC2015-46086.

Tseng, K. C., Thi, D., Giau, N. and Huang, P. (2017) ‘A Systematic Review of the Potential Application of Virtual Reality Within a User Pre-occupancy Evaluation’, in *UAHCI 2017: Universal Access in Human–Computer Interaction. Human and Technological Environments*, pp. 612–620. doi: 10.1007/978-3-319-58700-4.

Tumkor, S. and Esche, S. K. (2013) ‘Hand gestures in CAD systems’, in *Proceedings of the ASME 2013 International Mechanical Engineering Congress and Exposition*, pp. 1–9.

Vinayak, Murugappan, S., Liu, H. and Ramani, K. (2013) ‘Shape-It-Up: Hand gesture based creative expression of 3D shapes using intelligent generalized cylinders’, *CAD Computer Aided Design*. Elsevier Ltd, 45(2), pp. 277–287. doi: 10.1016/j.cad.2012.10.011.

Vizard (2015). Available at: http://www.worldviz.com/vizard-virtual-reality-software (Accessed: 27 July 2017).

VRPN (2016). Available at: https://en.wikipedia.org/wiki/VRPN (Accessed: 19 January 2016).

Wang, F. and Wang, Z. (2016) ‘Robust Features of Finger Regions Based Hand Gesture Recognition Using Kinect Sensor’, in Tan, T., Li, X., Chen, X., Zhou, J., Yang, J., and Cheng, H. (eds) *Pattern Recognition: 7th Chinese Conference, CCPR 2016, Proceedings, Part I*. Singapore: Springer Singapore, pp. 53–64. doi: 10.1007/978-981-10-3002-4_5.

Wang, R. Y. and Popović, J. (2009) ‘Real-time hand-tracking with a color glove’, *ACM Transactions on Graphics*, 28(3), p. 1. doi: 10.1145/1576246.1531369.

Wang, Z. B., Ong, S. K. and Nee, a. Y. C. (2013) ‘Augmented reality aided interactive manual assembly design’, *International Journal of Advanced Manufacturing Technology*, 69(5–8), pp. 1311–1321. doi: 10.1007/s00170-013-5091-x.

Weichert, F., Bachmann, D., Rudak, B. and Fisseler, D. (2013) ‘Analysis of the accuracy and robustness of the Leap Motion Controller’, *Sensors (Switzerland)*, 13(5), pp. 6380–6393. doi: 10.3390/s130506380.

Wingrave, C. A., Williamson, B., Varcholik, P. D., Rose, J., Miller, A., Charbonneau, E., Bott, J. and Laviola, J. J. (2010) ‘The wiimote and beyond: Spatially convenient devices for 3D user interfaces’, *IEEE Computer Graphics and Applications*, 30(2), pp. 71–85. doi: 10.1109/MCG.2009.109.

Wobbrock, J. O., Wilson, A. D. and Li, Y. (2007) ‘Gestures without libraries, toolkits or training: a 1 recognizer for user interface prototypes’, in *Proceedings of the 20th annual ACM symposium on User interface software and technology UIST 07*, pp. 159–168. doi: 10.1145/1294211.1294238.

Xu, C. Bin, Zhou, M. Q., Shen, J. C., Luo, Y. L. and Wu, Z. K. (2015) ‘A Leap Motion based intuitive volume interaction technology’, *Journal of Electronics and Information Technology*, 37(2), pp. 353–359. doi: 10.11999/JEIT140370.

Yeung, L. F., Cheng, K. C., Fong, C. H., Lee, W. C. C. and Tong, K.-Y. (2014) ‘Evaluation of the Microsoft Kinect as a clinical assessment tool of body sway.’, *Gait & posture*. Elsevier B.V., 40(4), pp. 532–8. doi: 10.1016/j.gaitpost.2014.06.012.

Zhang, Z. (2012) ‘Microsoft kinect sensor and its effect’, *IEEE Multimedia*, 19(2), pp. 4–10. doi: 10.1109/MMUL.2012.24.

Appendix I

The Evaluation of Gesture-based Design Review System

In this work, we propose a body gesture-based design review system for CAD models to be reviewed in a virtual environment. We want to evaluate our system by comparing the existing keyboard and mouse CAD system with our system. Six commands: translation, scaling, rotation, navigation, exploding, assembly are designed for the design review process.

1. Age

____

2. Have you used CAD software before? (such as: CATIA, SOLIDWORKS, AUTOCAD, UG)

○ Yes ○ No

3. Is it easy or difficult to learn and remember the designed gestures in the training process? (please give you mark according to the easiness)

○ 1 (easy) ○ 2 ○ 3 (moderate) ○ 4 ○ 5 (difficult)

4. Is it easy or difficult to understand the meaning of the designed body gestures? (please give your mark according the easiness)

○ 1 (easy) ○ 2 ○ 3 (moderate) ○ 4 ○ 5 (difficult)

5. Is it easy or difficult to learn and remember CAD command in professional CAD software using mouse and keyboard? (please give your mark)

○ 1 (easy) ○ 2 ○ 3 (moderate) ○ 4 ○ 5 (difficult)

6. Without considering the exact function provided, taking naturalness and intuitiveness into consideration, what do you think of the current CAD software interaction with the computer using the mouse and keyboard?

○ awkward ○ boring ○ acceptable ○ good ○ perfect

7. Only considering the interaction, what do you think of the new gesture-based design review system?

○ awkward ○ boring ○ acceptable ○ good ○ perfect

8. Do you think the designed gesture can intuitively represent the CAD command for the design review system and it is easy to remember the meaning of gestures during the training session?

○ 1 (totally cannot remember) ○ 2 (remember a little but less than half)

○ 3 (remember half of them) ○ 4 (remember most of them)

○ 5 (remember all of them)

9. If only considering the proposed functions, which interaction way do you prefer to use ? (the computer mouse and keyboard or body gesture)

○ mouse and keyboard ○ body gesture

Appendix II

The Evaluation of the Hand Gesture-Based Design Review System

1. Name

____

2. Age

____

3. Do you have any experience with CAD software? (SolidWorks, Catia, ProE, etc)

○ Yes

○ No

4. The learning time of using the computer mouse and keyboard-based CAD system. (if you have experience with CAD software before, just review how long it took to learn the CAD software comparing the gesture-based software) (1-10, 1:short, 10:long)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

5. Intuitiveness of the mouse and keyboard-based design review system. (1: bad, 10: good) (Intuitiveness means the command is easy to remember)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 □○ 6 ○ 7 ○ 8 ○ 9 ○ 10

6. Naturalness of the mouse and keyboard-based design review system. (1: bad, 10: good) (naturalness means simple)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

7. The cognitive load of the mouse and keyboard-based design review system. (1: low, 10: high) (cognitive means memory load)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

8. Ergonomic comfort of using the mouse and keyboard during the design review. (1: bad, 10: good) (ergonomic comfort means easiness to perform)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

9. The learning time of the gesture-based CAD system. (1-10, 1:short, 10:long)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

10. Intuitiveness of the gesture-based design review system. (1: bad, 10: good) (Intuitiveness means the command is easy to remember)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

11. Naturalness of the gesture-based design review system. (1: bad, 10: good) (naturalness means simple)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

12. Cognitive load of the gesture-based design review system. (1: low, 10: high) (cognitive means memory load)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 □○ 10

13. Ergonomic comfort of using gestures during the design review. (1: bad, 10: good) (ergonomic comfort means easiness to perform)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10

Appendix III

User Experience Questionnaire for the design review system

Number __________ Design review system __________

(1: Body gesture, 2:Hand gesture, 3: Body and hand gesture)

The UEQ contains 6 scales with 26 items:

Attractiveness*:* The overall impression of the system.

Perspicuity*:* The easiness to get familiar with the system and learn how to use the system.

Efficiency*:* The users can review the design model thoroughly without any unnecessary effort.

Dependability*:* The users feel in control of the human computer interaction.

Stimulation*:* The users feel exciting and motivating to use the system.

Novelty*:* The system is innovative and creative and it can catch users' interest.

An example of an item is:

Attractive		Unattractive

The items are scaled from 1 to 7. Thus, 1 represents the most negative answer, 4 a neutral answer, and 7 the most positive answer.

Attractiveness			Stimulation		
Annoying		Enjoyable	Inferior		Valuable
Bad		Good	Boring		Exciting
Unlikable		Pleasing	Not interesting		Interesting
Unpleasant		Pleasant	Demotivating		Motivating
Unattractive		Attractive	**Dependability**		
Unfriendly		Friendly	Unpredictable		Predictable
Efficiency			Obstructive		Supportive
Slow		Fast	Not secure		Secure
Inefficient		Efficient	Does not meet expectations		Meet expectations
Impractical		Practical	**Novelty**		
cluttered		Organized	Dull		Creative
Perspicuity			Conventional		Inventive
Not understandable		understandable	Usual		Leading edge
Difficult to learn		Easy to learn	Conservative		Innovative
Complicated		Easy			
Confusing		Clear			