21ST INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED17
21-25 AUGUST 2017, THE UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, CANADA

A HAND GESTURE-BASED INTERFACE FOR DESIGN REVIEW USING LEAP MOTION CONTROLLER

Xiao, Yu; Peng, Qingjin
University of Manitoba, Canada

Abstract

Computer-aided Design (CAD) has improved the original pencil and paper-based design drawing method. Designers can modify their designs using the mouse and keyboard as input devices of CAD commands. But there is a lack of design tools using a natural interface with an intuitive way for the design input. A hand gesture-based design interface is proposed in this paper for the review of CAD models in virtual environments. Leap Motion Controller is used to capture hand gestures of users to operate CAD models and model components. A template mapping method is applied in the gesture recognition for the manipulation of CAD models. Applications of the proposed interface show that the gesture input can effectively operate the model. User test results verify that the proposed system provides a natural and intuitive user experience in the CAD model manipulation compared to the traditional input method using the computer mouse and keyboard.

Keywords: Computer Aided Design (CAD), Communication, Visualisation, Hand gesture

Contact:
Prof. Qingjin Peng
University of Manitoba
Mechanical Engineering
Canada
qingjin.peng@umanitoba.ca

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 8: Human Behaviour in Design, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Many research activities in the human-computer interaction (HCI) have been conducted in the design field. Designers can 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. Availability of the low cost, non-contact, vision-based tracking devices has grown interests. Microsoft launched Kinect in 2010 to detect the human skeleton with a software development kit (SDK). Studies for the hand gesture recognition using depth data from the Kinect sensor mainly focus on static gestures (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, another sensor called Leap Motion Controller (LMC) has been developed to track the hand motion (Leap Motion Controller, 2015). The LMC as a hand motion tracking device has 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 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 the user immersive feeling and depth perception of 3D objects. Therefore, using devices like the LMC via hand gestures as design input methods 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.

A hand gesture-based design interface is developed in this research using the Vizard VR software (Vizard, 2015) and 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 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. Captured data from the LMC are mapped with the gesture templates to obtain the user command to operate the model.

Following parts of the paper are organized as follows. The related work is discussed in the next section. Descriptions of the gesture design and mapping methods are then introduced, followed by the system implementation and test. Conclusions and the future work are discussed at the end of the paper.

2 RELATED WORK

The development of HCI devices has created new ways of the human-machine interaction and CAD model manipulations (Karolczak and Klepaczko, 2014). Kumar et al. (2012) applied the data glove for painting and writing characters in a real-time environment. Stoerring et al. (2004) utilized a head mounted device (HMD) and head mounted camera (HMC) for the gesture recognition. The WII remote controller (Wingrave et al., 2010) and wrist-worn gloveless sensor (Kim et al., 2012) were also applied in HCI. After the release of the LMC as a new contact-free input device for the gesture-based HCI, its performances have been analysed in different research.

Apostolellis et al. (2014) evaluated the computer mouse and LMC in performing manipulation tasks for a stage light application. They designed an experiment to test the performance of these two devices for switching a virtual light on and off. The computer mouse used two steps in performing a 3D translation task, whereas the LMC only needed one step. The computer mouse performed better in the completion time and manipulation accuracy, and the LMC showed a better intuitiveness. Regenbrecht et al. (2013) presented a LMC supported hybrid augmented reality (AR) interface. They developed an interface using the LMC to track users' fingers with a webcam for an augmented view.

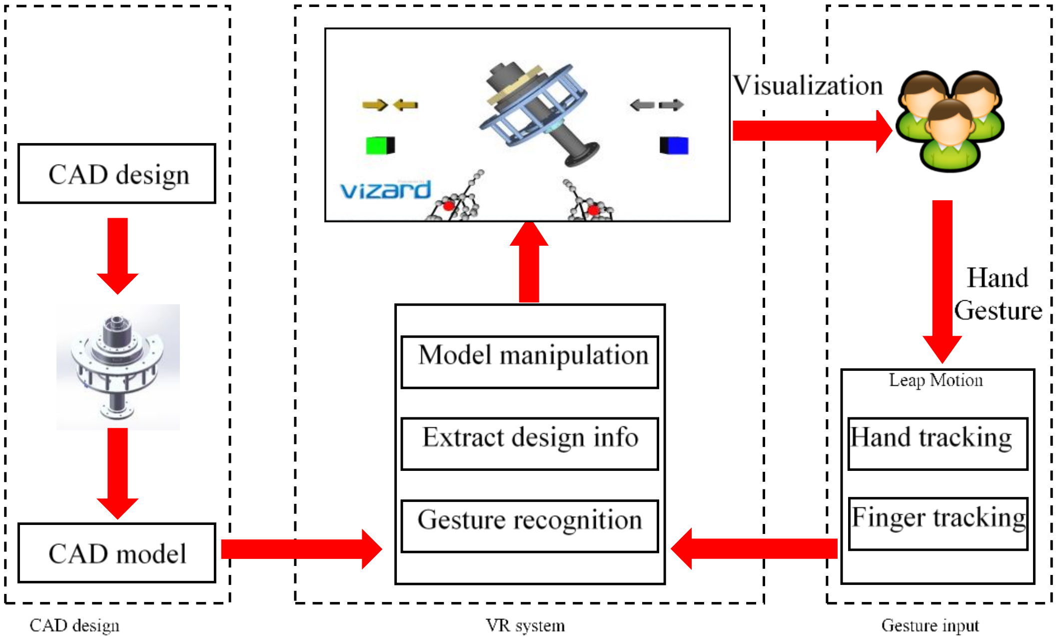


Figure 1. System structure

Xu et al. (2015) proposed a non-touch volume interaction prototype using the LMC. They proposed a 3D volume interactive interface for the medical image visualized in different layers. Karolczak and Klepaczko (2014) presented an innovative 3D viewer for magnetic resonance angiography (MRA) images. They applied the non-contact navigation and stereoscopic 3D vision technique in the vessel segmentation for 3D magnetic resonance angiography images, the LMC was used for navigating vascular structures and manipulating 3D models. Seixas et al. (2015) presented an experimental study on two selection gestures of the hand grab and screen tap using the LMC in 2D pointing tasks. They used the ISO 9241-9 multi-directional tapping test in the gesture analysis of the error rate and accuracy. Their results indicate that the hand grab gesture performed better than the screen tap. Bachmann et al. (2015) conducted a study to compare the computer mouse and LMC in selection tasks based on a Fitts' law-based analysis for users' performance. Kerefeyn and Maleshkov (2015) developed a gesture-based approach for manipulating virtual objects using the LMC.

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) and Hidden Markov Models (HMM) (Mccartney et al., 2015). 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).

Most of the gesture recognition methods use the colour information of hands and image processing techniques to extract colour data of hands for a special gesture design. However, the light condition in the environment affects the quality of images processed, which may cause negative effects for extracting hand gestures. The LMC is explicitly targeted for hand motion tracking. The orientation of hands and the position of fingers are computed automatically. The image processing task is not mandatory to extract gestures. 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 trackers suggests that it can be an effective tool for detecting hand gestures (Guna et al., 2014). Integrating the LMC with VR systems seems a novel way to explore the CAD model. The contribution of this research is to integrate VR environments with a gesture-based interactive CAD design interface to provide users an intuitive and natural way of the design input for the model manipulation, and a template mapping method for the real-time hand gesture recognition.

3 GESTURE AND MAPPING

3.1 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 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 2.

Attributes in Table 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 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 we only map them with corresponding commands 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.

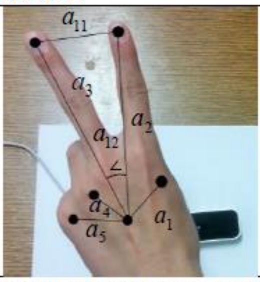
The scaling operation uses two index fingers moving apart from each other. The gesture acts as a trigger signal and adjust the zooming parameter. The user zooms in or out by widening or narrowing the distance between two index fingers.

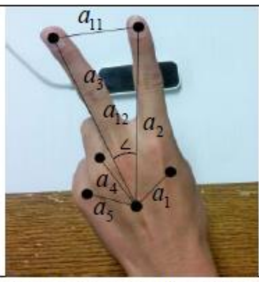
Table 1. CAD commands and proposed gestures

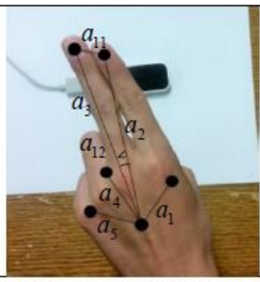
Software	**CAD commands**	**Operations**	**Functions**	**Gestures**
Solidworks	Pan	Translation	Translate the model	Translation
AutoCAD	3D pan	Translation		
CATIA	Pan	Translation		
Solidworks	Rotate	Rotation	Rotate the model	Rotation
Solidworks	Roll	Rotation		
AutoCAD	Rotate	Rotation		
AutoCAD	3D Rotate	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 2. Gesture definition

Translation

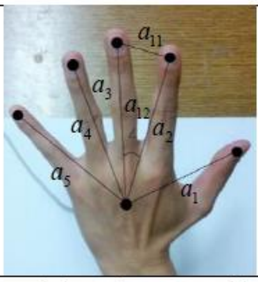


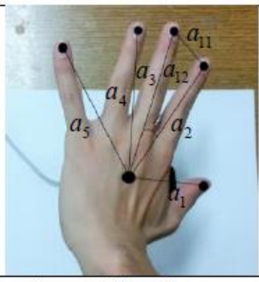


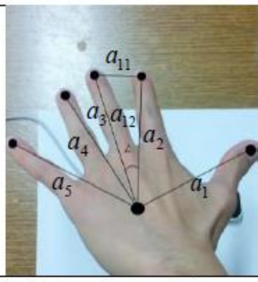


Index and middle fingers of the left hand perform pinch gestures for grab and release commands. When a model is grabbed, it can be translated to any place in the interaction window. The model is released by opening fingers. A template of the translation gesture is defined using joints of palm and thumb (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}).

Rotation

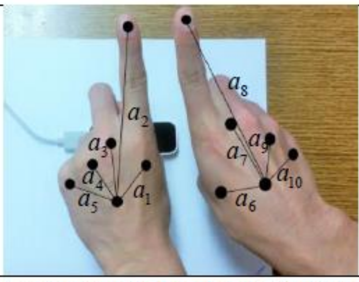


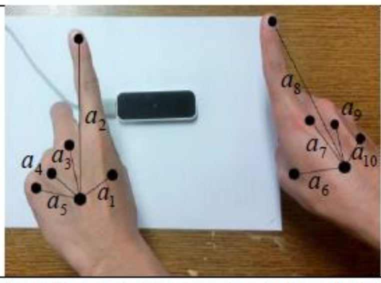




The model can be rotated based on the rotation of five fingers of the left hand. A template of the rotation gesture is defined using joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), index and middle (a_{11}) and the angular relationship between index and middle fingers (a_{12}).

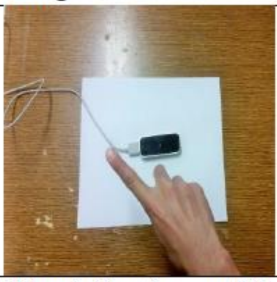
Scaling

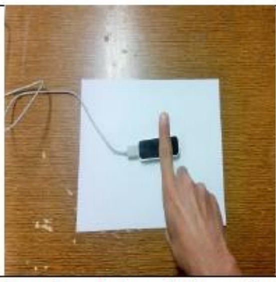


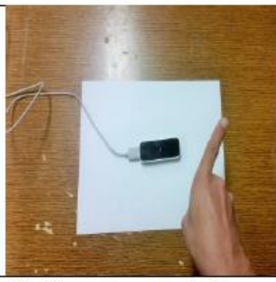


Two index fingers are used for scaling the model. A template of the scaling gesture is defined using left hand joints of palm and thumb (a_1), palm and index (a_2), palm and middle (a_3), palm and ring (a_4), palm and pinky (a_5), and right hand joints of palm and thumb (a_6), palm and index (a_7), palm and middle (a_8), palm and ring (a_9), palm and pinky (a_{10}).

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 original model.

Table 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

3.2 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 captured. Gesture mapping is to match the captured data to the template. A flow chart of the gesture mapping process is shown in Figure 2.

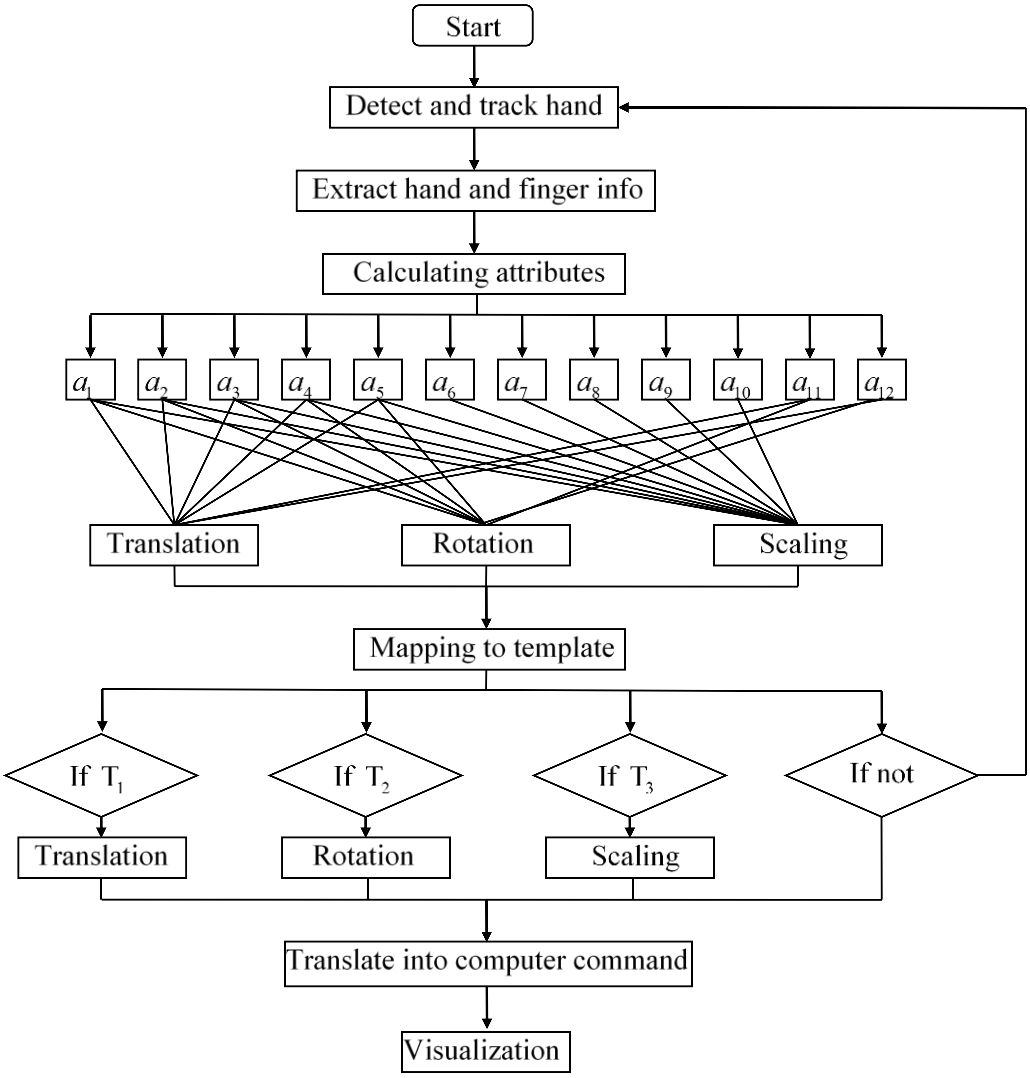


Figure 2. Gesture mapping flow chart

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:

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

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. 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 3. Based on Equation 1, as the result of $g(T_2,\ D)$ equals to 1, the rotation gesture is recognized.

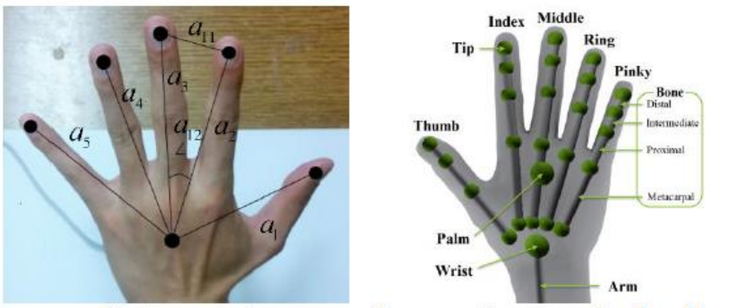
Table 4. Gesture mapping example

F \ P	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

A \ O	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 3. Example of the gesture mapping (Leap Motion Controller, 2015)

4 IMPLEMENTATION AND APPLICATIONS

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 is applied as an example 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, 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. In the manipulation process shown in Figure 4, a) - f) show the manipulation of detailed models, g) - i) display the manipulation of the complete model.

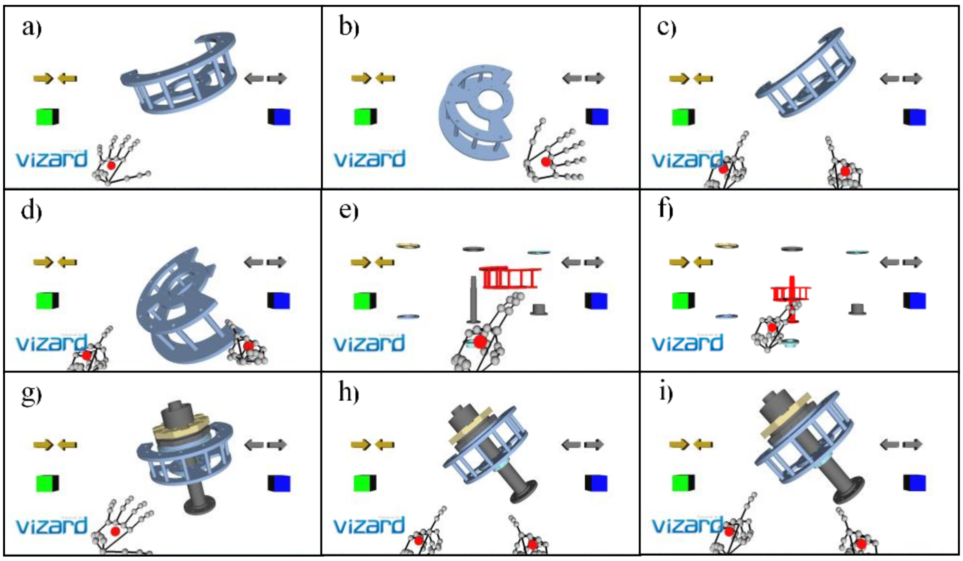


Figure 4. Design review interface

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 5. The recognition rates of the correct gestures are over 80%.

Table 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%

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 the scale value in the questionnaire (Guerra, Gidel and Vezzetti, 2016). The statistical results are shown in Table. 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. 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	

5 CONCLUSIONS AND FUTURE WORK

This paper presented a hand gesture-based human-computer interface to provide users a natural and intuitive way of manipulating CAD models. The Leap Motion Controller is used for the hand tracking and gesture recognition. The Vizard system is utilized as a virtual environment to develop the user interface. The user survey showed users' preference of using hand gestures rather than the mouse and keyboard as input methods for the design review. The interface provides flexible operations for non-experienced CAD users. The common CAD commands are included in the gesture operations (Song et al., 2014). This system uses a low-cost device that is affordable for most of users. Using the Leap Motion Controller with infrared (IR) imaging for tracking gestures, users do not have to wear any sensor or equipment to move in the interaction area. For the future work, a comparison of the proposed system with others approaches for the computer input such as the WII remote controller and game joystick will be conducted. More users will be recruited for the user test to improve the system. Additional commands will be added to the gestures to fulfil requirements of both viewing and modifying product models.

REFERENCES

Apostolellis, P., Bortz, B., Polys, N., Peng, M. and Hoegh, A. (2014), "Exploring the integrity and separability of the Leap Motion Controller for direct manipulation 3D interaction", *IEEE Symposium on 3D User Interfaces*, Minneapolis, USA, 29-30 March 2014, IEEE Xplore, pp. 153-154. https://doi.org/10.1109/3dui.2014.6798866.

Bachmann, D., Weichert, F. and Rinkenauer, G. (2015), "Evaluation of the leap motion controller as a new contact-free pointing device", *Sensors*, Vol.15, No.1, pp. 214-233. https://doi.org/10.3390/s150100214.

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", *14th International Design Conference, Cavtat-Dubrovnik Croatia*, 16-19 May 2016, Design, Design theory and research methods, 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*, Vol.14 No.2, pp. 3702-3720. https://doi.org/10.3390/s140203702.

Karolczak, K. and Klepaczko, A. (2014), "A stereoscopic viewer of the results of vessel segmentation in 3D magnetic resonance angiography images", *Software Practice Advancement Conference*, London UK, 29 June-2 July 2014, Proceedings of the 2014 SPA.

Kerefeyn, S. and Maleshkov, S. (2015), "Manipulation of virtual objects through a LeapMotion optical sensor", *International Journal of Computer Science Issues*, Vol.12 No.5, pp. 52-57.

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", *The 25th annual ACM symposium on User interface software and technology*, Cambridge USA, 07-10 Oct 2012, Proceedings of the 25th annual ACM symposium on User interface software and technology, pp. 167-176. https://doi.org/10.1145/2380116.2380139.

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*, Vol. 43, pp. 15-26.
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*, Vol.4 No.3, pp. 176-179.
Leap Motion (2015). Available at: https://www.leapmotion.com/.
Marin, G., Dominio, F. and Zanuttigh, P. (2014), “Hand gesture recognition with leap motion and kinect devices”, *IEEE International Conference on Image Processing*, Paris France, 27-30 Oct 2014, IEEE Xplore, pp. 1565-1569. https://doi.org/10.1109/icip.2014.7025313.
Mccartney, R., Yuan, J. and Bischof, H. (2015), “Gesture Recognition with the Leap Motion Controller”, *The 19th International Conference on Image Processing, Computer Vision, and Pattern Recognition*, Las Vegas USA, 27-30 July 2015, Proceedings of IPCV'15, pp. 3-9.
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*, Vol.5 No.7, pp. 476-482.
Nguyen-Dinh, L. Van, Roggen, D., Calatroni, A. and Tröster, G. (2012), “Improving online gesture recognition with template matching methods in accelerometer data”, *The 12th International Conference on Intelligent Systems Design and Applications*, Kochi India, 27-29 Nov 2012, IEEE Xplore, pp. 831-836. https://doi.org/10.1109/isda.2012.6416645.
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*, Vol.3 No.3, pp. 2003-2010.
Regenbrecht, H., Collins, J. and Hoermann, S. (2013), “A Leap-supported, hybrid AR interface approach”, *The 25th Australian Computer-Human Interaction Conference on Augmentation, Application, Innovation, Collaboration*, Adelaide Australia, 25-29 Nov 2013, pp. 281-284. https://doi.org/10.1145/2541016.2541053.
Seixas, M. C. B., Cardoso, J. C. S. and Dias, M. T. G. (2015), “One Hand or Two Hands? 2D Selection Tasks With the Leap Motion Device”, *The Eighth International Conference on Advances in Computer-Human Interactions*, Lisbon Portugal, 22-27 Feb 2015, pp. 33-38.
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*, Vol.46, pp. 239-245. https://doi.org/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”, *International Conference on Visualization, Imaging and Image Processing*, Marbella Spain, Sep, Proceeding of the 4th IASTED, pp. 766-771.
Thakur, A. and Rai, R. (2015), “User Study of Hand Gestures for Gesture Based 3D Cad Modeling”, *ASME International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Boston USA, 2-5 Aug 2015, https://doi.org/10.1115/detc2015-46086.
Vinayak, Murugappan, S., Liu, H. and Ramani, K. (2013), “Shape-It-Up: Hand gesture based creative expression of 3D shapes using intelligent generalized cylinders”, *Computer Aided Design*, Vol.45 No.2, pp. 277-287.
Vizard (2015). Available at: http://www.worldviz.com/vizard-virtual-reality-software/.
Weichert, F., Bachmann, D., Rudak, B. and Fisseler, D. (2013), “Analysis of the accuracy and robustness of the Leap Motion Controller”, *Sensors*, Vol.13 No.5, pp. 6380-6393. https://doi.org/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*, Vol.30 No.2, pp. 71-85. https://doi.org/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”, *The 20th annual ACM symposium on User interface software and technology UIST 07*, Newport USA, 07-10 Oct 2007, pp. 159-168. https://doi.org/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*, Vol.37 No.2, pp. 353-359.

ACKNOWLEDGMENTS

This research has been supported by the Discovery Grants (RGPIN-2015-04173) from the Natural Sciences and Engineering Research Council (NSERC) of Canada, and by the Graduate Enhancement of Tri-Council Stipends (GETS) program from the University of Manitoba.