Interactive Hand Gesture-based Assembly for Augmented Reality Applications

Rafael Radkowski
Heinz Nixdorf Institute, Product Engineering
University of Paderborn
Paderborn, Germany
rafael.radkowski@hni.uni-paderborn.de

Christian Stritzke
University of Paderborn
Paderborn, Germany
cstritzk@uni-paderborn.de

***Abstract*—This paper presents an Augmented Reality (AR) assembly system for the interactive assembly of 3D models of technical systems. We use a hand tracking and hand gesture recognition system to detect the interaction of the user. The Microsoft Kinect video camera is the basis. The Kinect observes both hands of a user and the interactions. Thus, a user can select, manipulate, and assemble 3D models of mechanical systems. The paper presents the AR system and the interaction techniques we utilize for the virtual assembly. The interaction techniques have been tested by a group of users. The test results are explained and show that the interaction techniques facilitate an intuitive assembly.**

Keywords - Augmented Reality; Interaction; Interactive Assembly

I. INTRODUCTION

The Augmented Reality (AR) technology is a kind of human-computer interaction that superimposes the visual perception of a user with computer-generated information (e.g., 3D models, annotations, and texts). AR presents this information in a context-sensitive way to the user. Special viewing devices are necessary to use AR. A common viewing device is the so-called head mounted display (HMD); a device similar to eyeglasses that use small displays instead of lenses. In the field of mechanical engineering, AR applications are explored for assembly, service, maintenance, and design reviews [1] [2].

Intuitive interaction techniques are a major aspect of AR applications. Intuitive means, a user is able to manipulate virtual objects without the aware utilization of prior knowledge. For that purpose, different interaction techniques and concepts have emerged in the field of AR.

A major driver for intuitive interaction is vision-based hand gesture recognition. Computer vision (CV) algorithms analyze a video stream, detect the user's hands, and determine a gesture. The gesture fires an interaction function of the AR application that manipulates (e.g., translate, etc.) a virtual object.

Hand gestures and computer vision-based hand gesture recognition are assumed to approximate the natural interaction of humans closely [3]. A user can interact with a virtual object like s/he interacts with a physical object using his/her hands. One advantage is, that a user does not need to wear or carry any technical device in his/her hand.

However, hand gestures and hand gestures recognition is still a challenging research field. In the field of AR, the techniques have not left the research laboratories until today. One reason is the need of technical devices that are attached to the user's hand in order to track it. The user also still act as operator of a machine; the interaction is not intuitive [4].

New types of video cameras like Microsoft Kinect facilitate the free-hand interaction. This video camera, particularly its images are utilized to detect the hands of a user and the hand gesture. However, it is necessary to explore interaction techniques and interaction metaphors for the interaction with virtual objects.

This paper presents an AR assembly system that facilitates the virtual assembly of virtual parts. A Kinect video camera observes the user and its interactions; the position of the hands and their gestures are recognized. No addition devices need to be attached to the hands of the user. This paper explains the system and describes the interaction metaphors, which are suitable for a free-hand interaction. The metaphors have been tested by a group of users. The results are presented.

This paper is structured as following. In the next section, the relevant related work is reviewed. Afterwards the AR assembly system is introduced as well as the interaction metaphors for selection, manipulation, and the virtual assembly of 3D models. Section 4 describes the user test. The paper closes with a summary and an outlook.

II. RELEATED WORK

The related work addresses the field of hand gesture-based interaction in AR applications.

One aim of many AR applications is to provide a natural and intuitive interaction interface to the user [3] [5] [6]. Like physical objects, a user should be able to grasp a virtual object like a real object. The user should grasp an object with his/her fingers, pick it up, and place it at every desired location. Therefore, different approaches exist.

One of the first systems is introduced by Buchmann et al. [3]. Their system, called FingARtips, is a gesture-based system for the direct manipulation of virtual objects. They attached fiducial markers on each finger to track the fingertips and to recognize the hand gesture. This solution allows to pick up virtual objects. However, it uses markers.

Reifinger et al. introduce a similar system [5]. The authors utilize an infrared tracking system for hand and gesture recognition. Infrared markers are attached to the fingertips and hands of a user. A tracking system detects the markers and a computer-internal hand model is build using this data. Thus, the user is able to grasp virtual objects like real objects.

 ISBN: 978-1-61208-177-9

Lee et al. introduces a system that does not utilize physical markers [6]. The authors use a feature-based tracking system. Computer vision algorithms identify hand features. Thus, the system detects and tracks the user's hand. This allows a user to attach a 3D model virtually to his/her hand, to move it, and to place it on different positions. However, a realistic grasping action is not been realized.

Siegl et al. introduces the concept of 3D-cursors as interaction metaphor in AR applications [7]. The user of an AR application is able to indicate a point in space with his/her hand. A vision-based system recognizes the hand. This way, the indicated position is calculated.

We introduce our own hand recognition system in [8]. In this previous work, we investigate how important the visibility of the user's hand for interaction purposes is. Furthermore, the hand recognition system is introduced. Further systems can be found in [9], [10], [11], and [12]. These examples provide only an overview about the research field; it would exceed the size of this work to present all systems. However, two findings can be stated: first, intuitive, natural interaction is one aim of AR. However, the working systems utilized physical markers. Second, there are many efforts to realize a computer-vision based hand gesture recognition system that works without any physical markers. A practical solution does not exist until today.

III. AUGMENTED REALITY ASSEMBLY SYSTEM

This section describes the AR assembly system for the virtual assembly of virtual parts of mechanical systems. The provided interaction techniques facilitate to manipulate and to assemble virtual parts. All interactions are carried out by hand movements and hand gestures, without any devices attached to the user's hand. The section starts with an overview of the AR system, which includes a presentation of the used hardware and software. Afterwards the interaction techniques for selection and manipulation tasks are introduced as well as the interaction for the virtual assembly.

A. Overview

Figure 1 shows an overview of the hardware setup of the AR application. A table is the main working area. The user stands on a fixed position in front of the table. We use a monitor-based AR-system. It consists of a 24" widescreen monitor and a video camera. The video camera captures images of the scene in front of the user. It is located next to the user, close to the user's head. It simulates a camera attached on a head mounted display. We have decided for a statically arranged camera in order to get comparable test conditions. It is a Creative Live Cam Video IM Ultra webcam (1280 x 960 pixel at 30 fps). The user observes the augmented scene on the screen of the monitor. For tracking the ARToolKit is used, a pattern-based tracking system [13]. Altogether, the setup represents a common monitor-based AR application.

The Kinect video camera stands opposite to the user. It observes the user and the user's interactions. The camera is aligned into the direction of the user. It captures RGB color images with a resolution of 640 x 480 pixel and 12bit depth images. The user does not see these images during s/he uses the AR application. The working area of the Kinect camera is arranged manually to the working area of the user. Therefore, the user put his hand to two corners of a control image. A region of interest is specified with respect to these corners. Thus, the camera as well as the user has to stand on a fixed position after alignment.

As computer we used a PC with an Intel Xeon processor, ,3.5 GHz, 6GB RAM, and a NVIDIA Quadro 5000 graphics processor.

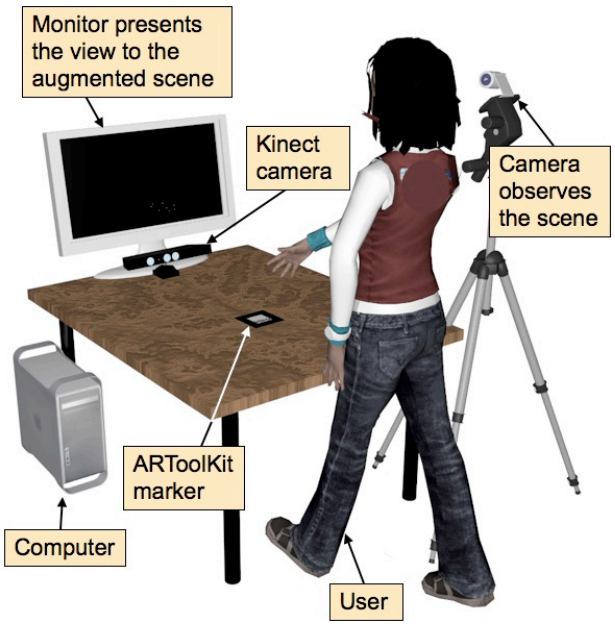

Figure 1. Overview of the hardware setup of the AR system

Two software tools are used to run the AR assembly system: a hand gesture recognition software and an AR application. The hand gesture recognition software detects the hand position in space and the hand gesture. It detects five gestures: fist, open hand, closed hand, index finger, and a waving gesture. The software is based on OpenCV (http://opencv.willowgarage.com/), an open source computer vision library.

The hand position and gesture are submitted as messages via UDP/IP to the second tool, the AR application. The AR application is based on OpenSceneGraph, an open source scene graph library (www.openscenegraph.org). It facilitates the rendering of 3D models, provides functions for collision detection and supports interaction. For our purpose, the AR application provides the following functions: selection of 3D models, translation, rotation, scale, change of attributes, and virtual assembly. The gesture recognition software and the AR application have been described in detail in [14].

B. Interactive Selection

Figure 2 shows the view of the user that is presented on the screen. It displays the main view of the application. The main view shows the virtual parts, which need to be assembled. In addition, multiple virtual button icons are shown. The button icons allow selecting a distinct function (i.e., translation, rotation, etc.).

 ISBN: 978-1-61208-177-9

The interactive selection allows a user to select a 3D model or a manipulation function. The main interaction object is a virtual cursor; a 3D sphere (3D cursor). The 3D sphere indicates the position of the user's hand. It follows the movement of one hand in three dimensions. To select a 3D model the user has to move his/her hand, in particular the 3D sphere to a 3D model. The selection is implemented as collision detection between the sphere and a virtual part. This collision is considered as a selection when the user applies a fist gesture. When the hand is opened, the model is released. Thus, grasping is simulated. As visual clue, a selected 3D model is colored yellow, a selected menu item is colored red.

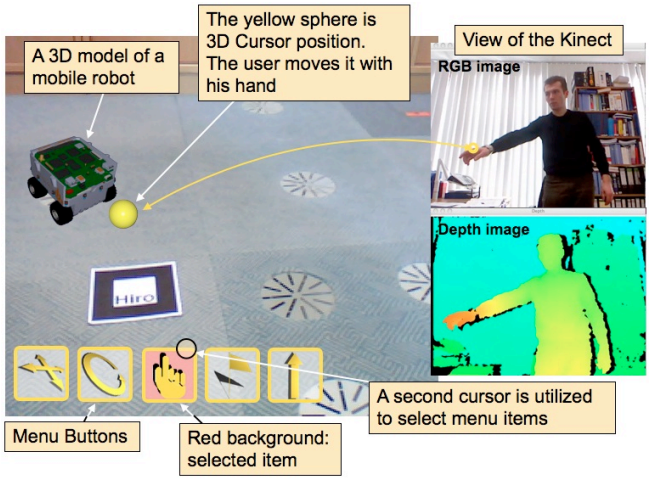

Figure 2. Main view of the AR application.

In addition to the 3D sphere, a 2D circle is used as selection object (also called 2D cursor). It is used to select a button icon in the menu. Usually, the 2D cursor is invisible. It appears only when the user moves his/her hand close to the button icons. Therefore, a region of interest (ROI) is specified that encloses the menu. If the hand of the user and the 3D sphere touches this ROI in screen coordinates the 2D cursor appears. To activate a function, the user has to move the circle above a button and has to wait for two seconds.

C. *Interactive Manipulation*

Interactive manipulation includes the translation, rotation, scaling, and the change of attributes (e.g. color) of a 3D model. The function can be applied to each 3D model after selection.

Two modes of operation exist: a so-called direct mode and a precise mode. The user can switch between both modes.

1) *Direct mode*

The direct mode allows to move and to rotate an object directly. The user can grasp it virtually and move it. Therefore, the user selects a 3D model. This 3D model is being attached to his/her hand and follows all movements. Furthermore, if the user rotates his/her hand, the model follows this rotation, limited to one degree of freedom.

The direct mode facilitates a fast movement of 3D models in the working area. In addition, this technique appears to be intuitive, because it meets a common grasping / pick & place operation. Unfortunately, it does not allow a precise alignment of virtual objects. Scaling and the change of attributes are also not supported in this mode.

2) *Precise Mode*

The precise mode facilitates an accurate translation, rotation, and scaling of virtual parts. 3D models are utilized as visual clues and interaction objects in order to support these functions.

Translation: Figure 3 shows the translation of a 3D model. The figure shows the view of the user. After selecting the function, a 3D model of a coordinate system appears above the virtual object that should be moved. This coordinate system indicates the possible moving directions.

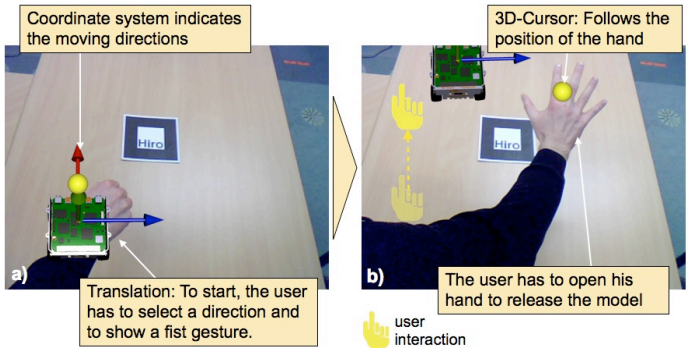

Figure 3. a) To translate a 3D model the user has to grasp an axis of the corrdinate system that appears above the 3D model, b) The user has to open his hand in order to release the 3D model.

The user has to choose one axis of the coordinate system (collision detection) using the yellow 3D cursor. To translate the model, a fist gesture has to be performed. This should simulate grasping. This starts the translation into the desired direction. To move the 3D model the user has to move his/her hand into the desired direction. The movement of the hand is assigned to the translation of the model. The user can select a scale factor that slows down or speeds up the movement. Usually, a scale factor between 0.5 and 1.5 is selected.

Rotation: After selecting this function and a 3D model, a virtual coordinate system appears above the 3D model. To rotate the object, the user has to grasp one axis of this coordinate system. Therefore, s/he uses the 3D cursor. To select the axis a fist gesture need to be performed. Doing this, the rotation starts. To rotate the 3D model the user rotates his/her hand about the selected axis. Every movement of the hand is transformed into a rotation. The start point is the angle, at which the fist gesture has been shown. It works similar for the rotation about the other axis. The interaction technique should simulate a rotation of an object using a lever. The lever is utilized to slow down or to magnify the rotation.

Scaling: The scaling works similar to the rotation and translation. After selecting this function and a 3D model, a virtual box and a coordinate system appears above the 3D model (Figure 4). To coordinate system indicates the

 ISBN: 978-1-61208-177-9

possible scaling directions. The box is a visual clue that helps to recognize the scaling factor. To scale the 3D model the user has to select an axis using the virtual 3D cursor and to perform a fist gesture. Then s/he has to move his/her hand along this axis. The movement is multiplied as scaling factor to the 3D model. In addition to the 3D model, the box is also scaled. If the 3D models are unshaped, the box facilitates to recognize the scaling factor.

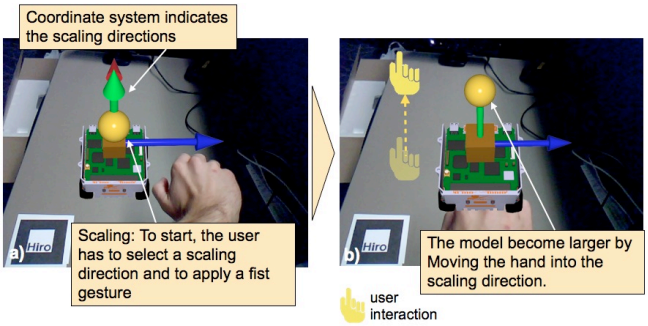

Figure 4. Scaling of 3D objects

D. *Interactive Virtual Assembly*

The main function of the virtual assembly system is to provide interactions for the assembly of virtual parts. Assembly means in this case that two or more models join together when they meet on a specified position. The assembly system utilizes the selection and manipulation functions presented before. In addition to these functions, a pre-defined mode switch simplifies the assembly of two models. In the following, the assembly is explained using an example of an axle and a ball bearing. It works also for all virtual parts in the same way.

The virtual assembly is based on a so-called port concept [15]. A port is a distinct position on the surface of a 3D model that is annotated by a joint. A joint limited the degrees of freedom between two virtual parts. An octahedron on the surface of the 3D model visualizes this joint (Figure 5). Five types of joints are implemented. Each type limits a different degree of freedom: hinge joint, ball & socket joint, linear bearing, rotation bearing, and a fixed joint. The different types are visualized by different colors of the octahedron. The joints and its position are specified in a pre-process.

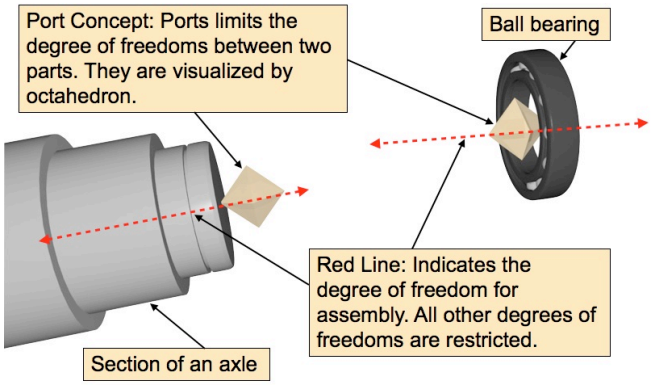

Figure 5. A port concept is used to assmble two virtual parts.

To assemble two parts the user has to move closely the two octahedrons of two parts. Two parts that using the same type of joint can be assembled only.

For the assembly task itself a two-step interaction is used [16]. Figure 6 shows the first step. The figure shows an axle and a ball bearing. Task of the user is to assemble the ball bearing wheel on an intended section of the axle.

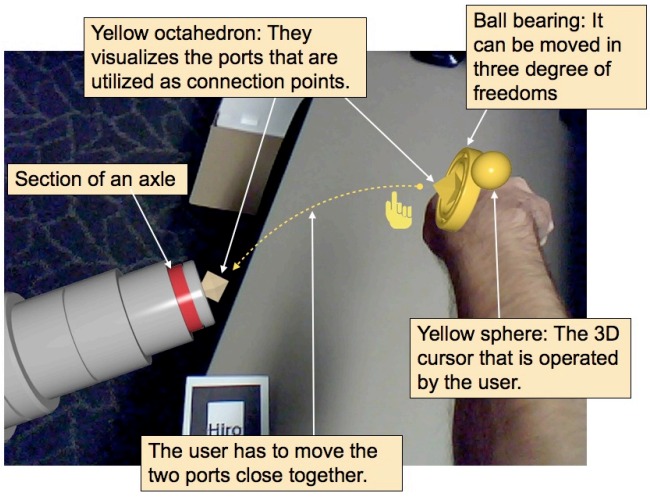

Figure 6. The assembly is carried out in two steps. In the first step, the user can move the 3D models in three degree of freedoms

In the first step, the user has to move both parts close together. Therefore, the mentioned direct mode can be used. Bounding boxes surrounds the octahedron. As soon the bounding boxes collide a mode switch is applied automatically. By this, all degrees of freedom of the two parts, which become fix after assembly, are being aligned. In this mode the user can move the part along the remaining degree of freedom only. In this example (Figure 7): as soon the ball bearing is close to the axle, the ball bearing is moved to the center axis of the axle. Furthermore, the ball bearing can be only moved along the center axis of the axle.

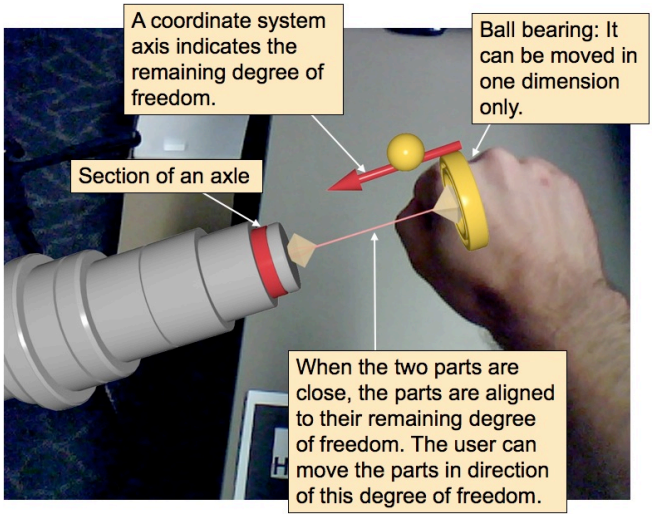

Figure 7. Second step of the virtual assembly. The user can only move the part in the remaining degree of freedom.

 ISBN: 978-1-61208-177-9

After the part is on the desired position, the user can release the part. Thus the ball bearing remains assembled on the axle and both parts can be moved as one group.

IV. USER TEST

The AR assembly system has been tested by a group of users. The aim of the test was to explore whether the interaction techniques facilitate the manipulation and the assembly of virtual parts. Furthermore, we wanted to discover whether the entire system meets the expectation of the users. We do not analysis the quality of the assembly, which would be necessary when using this AR application as simulation tool for real assemblies. In the following, the process is described and the results are presented. The section closes with a discussion of the results.

A. Process

During the test, the users should carry out an assembly task. Task was to assemble gear wheels, ball bearings, and clogging on an axle. The assembly tasks have been introduced to the test users by images. Each image has shown one assembly step (one part = one step). In summary, six parts have been needed to be assembled on the axle.

Before the test has started, the different interaction techniques were presented. Each user has gotten several minutes to practice the interaction techniques. During the test the user could decide on their own, which interaction techniques they want to use and what series of interactions are necessary to assemble two parts.

The test users were 15 students of the departments of mechanical engineering and computer science. No user has experience with hand gesture-based interaction techniques.

We have measured the time a user needed to assemble all parts. In addition, we have used a questionnaire to retrieve the opinion of the users. We asked eight questions (table 1). A Likert scala was used to rate the questions. The scale ranges from 1=”the statement meets my opinion” to 5=”I disagree with this statement” (The questionnaire and the answers were in German).

B. Results

Figure 8 presents the results of the time measurement with respect to the assembly task.

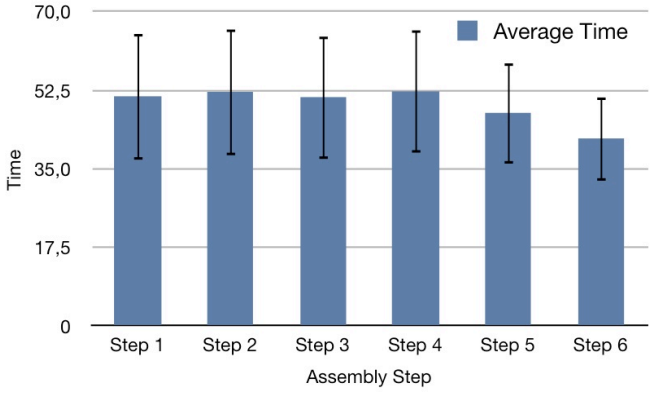

Figure 8. Average time per assembly time

The abscissa displays the six assembly steps, the ordinate displays the time. The bars indicate the average duration time for each particular step. The time measurement for each step has started automatically when two parts in the previous step were assembled. For the first step, the measurement was started manually. It can be observed that there are no significant changes between the different steps. However, there is a large variance.

Figure 9 presents the time measurement with respect to the interaction techniques. The abscissa shows the different interaction techniques, the ordinate the time. The bars show the average time each interaction technique has been in action. The time measurement has started when a user calls the function, it stops when a user has exit a function. The numerical values on each bar indicate the number each function has been called.

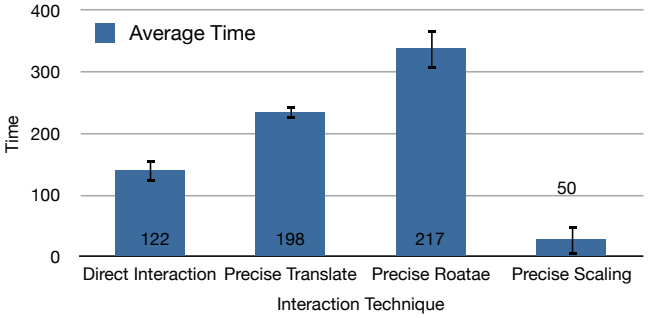

Figure 9. Average time per interaction technique

The results show that the precise interaction techniques demand more time than the direct manipulation techniques. The variance is marginal. More interesting is the number of function calls. It can be observed that the precise techniques have been called more often than the direct techniques.

Table 1 shows the results of the questionnaire. The first column contains the questions. The second column shows the average values of the answer and the third column the variance.

TABLE 1: *RESULTS OF THE QUESTIONNAIRE*

Question	Av. Result	Variance
1. I can use the manipulation interactions without any problems	1,53	0,52
2. It was easy to select the different 3D mdels	2,27	0,80
3. I understand the mode switch during the assembly task and was able to assemble the 3D models.	2,27	0,59
4. I used the direct mode for large movements and the precise mode to align objects	3,47	1,36
5. Overall, I had no problems to interact with 3D models.	1,93	0,46

C. Discussion

In general, the results of the user test prove that the selected interaction techniques facilitate the vitual assembly of virtual parts. All users were able to assemble the parts in six steps using the provided interaction techniques. All steps

 ISBN: 978-1-61208-177-9

could be completed. The duration of every step is nearly similar (Figure 8). The techniques appear to be controllable from the first time they have been used. A learning phase seems not to be necessary. Only at step 5 and step 6 a slow increase of the assembly time is recognizable. This likely indicates a learning effect, but it is not significant.

Figure 9 shows the average time for an interaction. The results show that the direct interaction is the fastest technique to translate a part. By putting time on the same level as difficulty, direct interaction is the simplest technique. The most difficult technique is the rotation. One reason is that the users need to grab the part several times. During the rotation, they lost the orientation and need to start over. The high number of rotation operations underpins this. The average time of precise scaling is low due to the fact that scaling was necessary in two assembly steps only.

However, there are some drawbacks. We assumed that the user uses the direct mode to move a part onto the workplace, close to the axle. Then s/he should use the precise mode to move the ports to a close distance before the mode switch and the two parts are aligned. The results show that a few users do not understand the intended way. The answer to question 3 of the questionnaire proves that a few users do not recognize this. In addition, the direct interaction was used 122 times. The low number results due to the fact, that six users do not use this technique (Figure 9).

In addition, the users have moved their hands very slow. They operate very carefully during the assembly task.

V. RESUMEE AND OUTLOOK

This paper presents a set of interaction techniques for the virtual assembly of virtual parts using a hand gesture-based interaction technique. We introduced a set of interaction techniques that allows interacting with virtual parts without using a graspable device. Therefore, we distinguish a direct mode and a precise mode. The direct mode allows fast translation. The precise mode facilitates a precise placing of virtual parts. In our opinion these two modes are necessary to facilitate an interaction with non-graspable parts. The user test gives us a strong indication that the techniques are capable to carry out a virtual assembly task. Finally the work shows that these separate techniques are a good choice for this kind of task.

The future work has two objectives. First, we will carry out an assembly using virtual and physical parts. Until now, virtual parts have been only used. This justifies no AR application. In the next step the users should be able to assemble virtual parts on physical parts.

Furthermore, we will test the precision of the Kinect and the entire AR system. Therefore, we will carry out an experiment with pick & place operations.

REFERENCES

[1] Radkowski, R., "What makes an Augmented Reality Design Review Successful?" in Horváth, I.; Mandorli, F.; Rusák, Z. (Eds.): Proceedings of the TMCE 2010 - Tools and Methods for Competitive Engineering, Ancona, Italy, April 12-16, 2010, pp. 499-510

[2] Azuma, R., "A Survey of Augmented Reality" In: In Presence: Teleoperators and Virtual Environments 6, 4 (August 1997), 1997, pp. 355-385

[3] Buchmann, V., Violich, S., Billinghurst, M., and Cockburn, A., "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 (GRAPHITE '04), 2004, pp. 212-221

[4] Malassiotis, S. and Strintzis, M.G., "Real-time hand posture recognition using range data," in Image and Vision Computing, 26(7), 2008, pp. 1027-1037

[5] Reifinger, S., Wallhoff, F., Ablassmeier, M., Poitschke, T., and Rigoll, G., "Static and Dynamic Hand-Gesture Recognition for Augmented Reality Applications," in Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments. Springer Verlag, Berlin / Heidelberg, 2007, pp. 728-737

[6] Lee, T. and Höllerer, T., "Hybrid Feature Tracking and User Interaction for Markerless Augmented Reality," in Proceedings of IEEE Virtual Reality 2008, 8-12 March, Reno, Nevada, USA, 2008, pp. 145-152

[7] Siegl, H., Schweighofer, G., and Pinz, A., "An AR Human Computer Interface for Object Localization in a Cognitive Vision Framework". In: Int. Workshop on Computer Vision in Human-Computer Interaction (ECCV), Springer 3058, 2004, pp. 176-186

[8] Radkowski, R. and Wassmann, H., "Using Computer Vision for Utilizing the Human Hand in An Augmented Reality Application". In: Proc. of IADIS Computer Graphics & Visualization, Lissabon, Portugal, 2007, pp. 127-131

[9] Hackenberg, G., McCall, R., and Broll, W., „Lightweight Palm and Finger Tracking for Real-Time 3D Gesture Control." in IEEE Virtual Reality 2011, pp. 19-26, Singapore, 2011

[10] Lu, Y. and Smith, S., "GPU-based Real-time Occlusion in a CAVE-based Augmented Reality Environment," in 2007 ASME International Design Engineering Technical Conference & Computers and Information in Engineering Conference (CIE), September 4 – 7, 2007 Las Vegas, Nevada, 2007, pp. 1131-1141

[11] Argyros, A.A. and Lourakis, M.I.A., "Tracking skin-colored objects in real- time," In: Cutting Edge Robotics, 2005, pp. 77–90

[12] Sung, K. K., Mi, Y., N., and Phill, K.R., "Color based hand and finger detection technology for user interaction," in International Conference on Convergence and Hybride Information Technology, 28-29 August 2008, pp. 229-236

[13] Kato, H. and Billinghurst, M., "Marker Tracking and HMD Calibration for a video-based Augmented Reality Conferencing System," in Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99), San Francisco, USA, 1999, pp. 85-94

[14] Radkowski, R. and Stritzke, C., „Comparison between 2D and 3D Hand Gesture Interaction for Augmented Reality Applications," in Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE 2010 Aug. 28-31, 2011, Washington, DC, USA, pp. 1-11

[15] Shen, Q., „A Method for Composing Virtual Prototypes of Mechatronic Systems in Virtual Environments," Ph.D. Thesis, University of Paderborn, HNI-Verlagsschriftenreihe, Paderborn, Vol. 193, Dez. 2006

[16] Vance, J. and Dumont, G., „A Conceptual Framework to Support Natural Interaction for Virtual Assembly Tasks," in Proc. Of the ASME 2011 World Conference of Innovative Virtual Reality, WINVR2011, Jun3 27-29, Milan, Italy, 2011, pp. 1-6

 ISBN: 978-1-61208-177-9