3D Gesture Interaction for Handheld Augmented Reality

Huidong Bai[1, 2*] Gun A. Lee[1†] Mukundan Ramakrishnan[2‡] Mark Billinghurst[1§]

The Human Interface Technology Laboratory New Zealand (HIT Lab NZ), University of Canterbury, New Zealand[1]
The Department of Computer Science, University of Canterbury, New Zealand[2]

Abstract

In this paper, we present a prototype for exploring natural gesture interaction with Handheld Augmented Reality (HAR) applications, using visual tracking based AR and freehand gesture based interaction detected by a depth camera. We evaluated this prototype in a user study comparing 3D gesture input methods with traditional touch-based techniques, using canonical manipulation tasks that are common in AR scenarios. We collected task performance data and user feedback via a usability questionnaire. The 3D gesture input methods were found to be slower, but the majority of the participants preferred them and gave them higher usability ratings. Being intuitive and natural was the most common feedback about the 3D freehand interface. We discuss implications of this research and directions for further work.

CR Categories: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities H.5.2 [Information Interfaces and Presentation]: User Interfaces—Interaction styles;

Keywords: handheld augmented reality, gesture interaction

1 Introduction

Handheld Augmented Reality (HAR) applications enable interaction with the user's physical environment through smartphones or tablets that superimpose virtual content on top of the real world. A variety of interaction techniques can be used to manipulate virtual objects in HAR applications. For example, tangible input with keypad or phone tilting can be used for translation and rotation control [Henrysson et al. 2007] [Chen et al. 2008], or direct touch screen input to select on-screen content [Langlotz et al. 2012] [Kim et al. 2011]. However, existing touch-based input methods have substantial limitations, such as finger occlusion of on-screen content, being limited to the physical screen size, and the difficulty of mapping two dimensional input into 3D spatial interaction [Mossel et al. 2013].

Since spatial interaction with a virtual scene is key for exploring the full potential of HAR, there is a need for research into richer and handheld interaction. Instead of physically touching the screen of mobile devices, gesture manipulation in midair has been one of the suggested alternative input methods for HAR systems. For example, with the gesture engine developed by Baldauf et al. [Baldauf et al. 2011], the user can select a virtual AR object by pointing at it, or grabbing, dragging and dropping it with a thumb and index finger pinch gesture on a smartphone with the help of the built-in mobile camera.

In our research we are interested in exploring how natural 3D gesture input can be used for one-handed interaction on HAR devices, overcoming some of the limitations of touch-based methods, and offering interactions with 3D AR objects in a more natural and intuitive way (Figure 1).

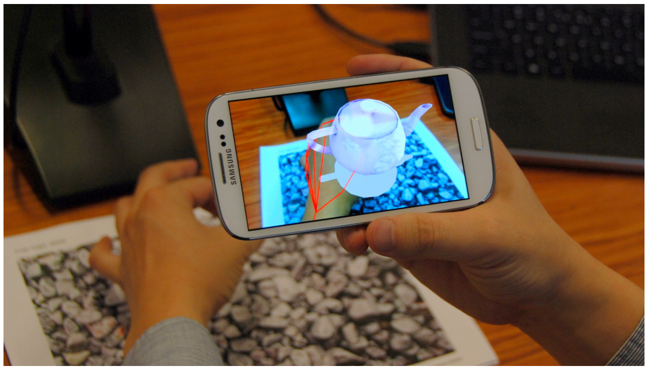

Figure 1: *The user is interacting with a virtual teapot using freehand input on the HAR device. A virtual hand skeleton is textured on the real hand image, and an augmented teapot is appearing overlaid on the tracked image.*

We organize this paper into five parts: first, we put our work in the context of related work, discussing the gesture-based HAR interfaces that have already been investigated. Next we present our new interaction technique, and explain the implementation of the system. Then we report on a formal user study that we conducted, detailing setup and procedure. After that we analyze the results and make a discussion. Finally, we conclude with an overall discussion and topic for future work.

2 Related Work

Researchers developed different methods to investigate gesture-based interfaces for HAR over recent years, each with its own advantages and limitations.

Early methods required the user to have a trackable marker attached on the fingers so the mobile camera can track their position. For instance, Henrysson et al. [Henrysson et al. 2007] attached a fiducial marker on the fingertip and used the mobile phone's front camera to capture its spatial position and pose for midair input. Hürst and Wezel [Hürst and Wezel 2013] investigated the potential for finger-based interaction by applying two color markers on two fingers respectively, and testing it in a mobile AR board game. Compared against the phone's touch screen and orientation sensors input, finger-based interaction produced the worst performance, but a high score of fun-engagement level that indicated the potential for gaming applications.

There have also been markerless gesture-based interfaces for HAR.

[*] e-mail: huidong.bai@pg.canterbury.ac.nz
[†] e-mail: gun.lee@hitlabnz.org
[‡] e-mail: mukundan@canterbury.ac.nz
[§] e-mail: mark.billinghurst@hitlabnz.org

SIGGRAPH Asia 2014, December 03 – 06, 2014, Shenzhen, China.
2014 Copyright held by the Owner/Author.
ACM 978-1-4503-1891-4/14/12
http://dx.doi.org/10.1145/2669062.2669073

Baldauf's gesture engine [Baldauf et al. 2011] detected the markerless finger pinch input based on skin-color segmentation and then assisted the user to manipulate virtual objects. Seo et al. [Seo et al. 2008] developed a whole hand input way for mobile AR interaction. Using a palm pose estimation method, a virtual object can be placed on the palm without needing visual markers, and can be moved along with the palm. These methods relied solely on skin color when segmenting the hand, making it difficult to continuously recognize the hand in an environment with skin-colored materials. Chun and Höllerer [Chun and Höllerer 2013] solved the skin color issue by using a trackable textured target as the background to subtract the hand region and detect the fingers, and then recognized a pinch motion for translation and scaling. However, the experiment results indicate that this type of hand interaction might not be a good fit for subtle hand motions yet.

Even though these papers presented the potential of using computer vision techniques to provide finger-based or palm-based gesture input on the mobile devices, one shortcoming was the lack of accurate depth sensing of hands, while full-fledged 3D interaction for mobile AR scenarios were not well investigated.

Bai et al. [Bai et al. 2013] built a HAR prototype on a tablet attached with an external depth camera via a standard USB cable connection. The movements of the user's fingertips in front of this camera were detected and mapped into corresponding manipulations of augmented virtual objects. Their study results indicated that the gesture interaction is more natural and enjoyable to use. They also reported on a prototype interface that supports natural 3D freehand gestures for wearable AR interaction [Bai et al. 2014]. They further explored combining hand gesture input with touch pad input that is commonly available on wearable computers. Their early pilot study results demonstrated that combining the natural gesture input with the touch pad improved both the performance and user experience significantly, which is more intuitive and natural while involved less mental stress while comparing with the touch input to common users.

In this paper, we investigate canonical manipulations based on 3D freehand interaction in popular one-handed HAR scenarios. We developed a framework to simulate future gesture-based input methods for smartphones. On a server PC we use a depth camera and gesture detection software to capture a 3D hand skeleton and identify fingertip positions. We then wirelessly transmit the gesture data to our smartphone, mapping midair gesture commands onto the selected virtual object shown on the mobile screen. Although depending on an external depth camera, this work prototypes the type of interactions that will be possible in the near future when depth sensing becomes widely available on HAR devices.

3 A HAR Framework for 3D Gesture Input

Although there are some research projects [Wang et al. 2011] [Kim et al. 2012] that have addressed full-fledged 3D hand tracking with the help of depth cameras or customized sensors, they are not feasible to be directly integrated into a mobile device because of complexity and heavy computational cost of gesture recognition algorithms. We investigate an alternative novel approach to implement 3D gesture interaction for handheld mobile AR based on a client-server framework. This would allow users to perform 3D manipulations in a one-handed HAR environment with their hand gestures in midair. We developed our prototype system combining three main technologies; (1) Three Gear Systems[1] hand tracking, (2) Vuforia[2] AR tracking, and (3) Alljoyn[3] wireless data communication.

We setup the whole system in a small workspace as shown in Figure 2. A PrimeSense[4] (RGB-Depth camera) is positioned 80 cm above the desired interaction space facing downwards, and a printed reference image marker is placed on the table right below the camera. A PC connected with the depth camera is used as a server, and a smartphone with a single built-in color camera is used as a client. The Three Gear software on the server uses the depth camera to generate on-the-fly 3D models of both hands in the space while using the image marker as a coordinate reference. Meanwhile, the Vuforia software on the phone enables the smartphone to track the position of the built-in camera relative to the same image marker. Thus, the two cameras use the same world coordinate system, the origin of which is located in the middle of the AR image marker, and there will be an established geometry transformation relationship between the server and the client camera views. The joint and fingertip positions of hands detected from the server is directly transmitted to the phone via a wireless network enabled by Alljoyn in real time. Thus the system eventually provides real-time 3D gesture interaction for video-based AR applications on handheld mobile devices.

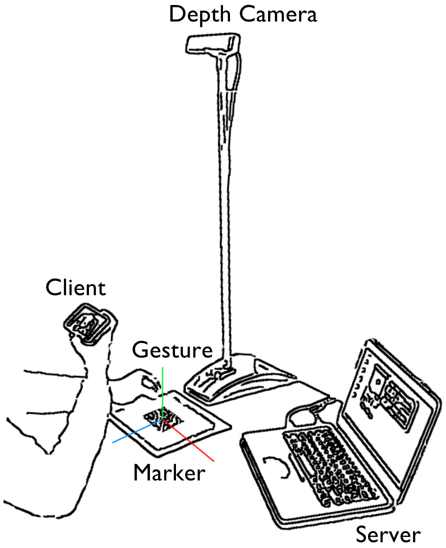

Figure 2: *System setup. The depth camera and the mobile camera share the same coordinate system of the AR marker. The hand gesture can be detected by the depth camera from the PC server side and the skeleton result can be wirelessly transmitted and mapped to AR scene on the phone client.*

The prototype system allows the user to manipulate virtual objects using pinch-based hand gestures which consists of pinching at a point of interest then moving the hand for further manipulation (similar to pressing a button on a mouse then dragging), and includes translation, rotation, and scaling. In the translation mode, the selected object follows the pinch position as the user moves the hand, while in the rotation mode it rotates according to the user hand orientation. Scaling is applied uniformly depending on the distance of the users hand displacement after pinching. In contrast, the traditional touch-based interaction utilizes screen touching instead of midair pinching to control the virtual target.

We run the server on a Dell laptop (Intel Dual-Core 2.4Ghz, 4GB RAM, Windows 7 OS) and the client on a Samsung Galaxy S3 smartphone (ARM Quad-core 1.4GHz, 1GB RAM, Android 4.1.2

[1] http://www.threegear.com/

[2] http://www.vuforia.com/

[3] http://www.alljoyn.org/

[4] http://www.primesense.com/

OS). The hand tracking runs at 25 frames per second (FPS) on the laptop with a fingertip accuracy of around 5 mm. In addition to tracking hand position and pose, the software can also recognize simple hand gestures such as finger pinching. Meanwhile, the AR tracking runs on the target phone at an interactive frame rate of around 30 FPS with 1280 by 720 pixel camera resolution while the communication delay between the server and the client averages less than 10 ms in our private local area network (up to 300 Mbps).

This framework allows us to rapidly prototype a HAR system that supports 3D hand gesture input. However, one limitation is that the user needs to keep their hands within the Three Gear detection volume (110 cm by 65 cm by 40 cm above the image marker in our setup), preventing them from moving outside the interactive space and losing the hand tracking. So while this is good for prototyping, in the near future we will need to develop a system running entirely on the handheld device when depth sensing becomes widely available on HAR devices.

4 User Study Design

We conducted a user study using a within-group design to investigate the benefits and drawbacks of our proposed gesture-based interface compared to the traditional touch-based interface. The main independent variable was the type of interface (touch and gesture) across three fundamental scenarios with varying tasks, and dependent variables included task performance and participants' ratings collected with preference questionnaires in terms of usability.

We implemented touch interaction to perform similar operations supported by gesture interaction. With touch interaction, users can switch between modes by three finger tapping, and simple tapping will select an object. Dragging selected objects will result in manipulating the object depending on the transformation mode. While one finger dragging only supports 2D transformation on the tracking plane, dragging with two fingers allows transformation along the axis perpendicular to the plane.

4.1 Experimental Environment and Task

We designed three experimental tasks in a HAR application each focusing on different canonical object manipulations: translation, rotation, and scaling. The first translation task requires the user to move a colorful cube in the lower left corner to overlap a half-transparent red cube located in the upper right corner in all three dimensions (Figure 3a), and the distance between them was 115 mm (130 mm by 80 mm on the marker plane) horizontally and 45 mm vertically. In the second task, a cube was placed with a rotation value of 310, -150, 10 degrees around local x, y, and z axis respectively. The user needs to rotate it back to a pose, in which the cube will align parallel to the axes on the side while having the blue side facing toward the user and pink side upward, as shown in Figure 3b. The third scenario needs the user to uniformly scale an inner colorful cube to approach the size of a half-transparent outer one (Figure 3c). The original edge size of the operational cube is 30 mm and the target dimension is 70 mm. It is noticeable that the touch-based scaling method was configured to the uniform scaling mode for keeping the comparison with the gesture input as fair as possible.

We asked users to perform each task by manipulating a virtual object to match the position, orientation, and size of a target object. Participants sat in front of the desk with a setup as presented in Figure 2, and they were asked to hold the smartphone with the non-dominant hand, so that they could use the dominant hand for touch input or for gesture interaction. They were asked to perform the task as quickly and accurately as possible.

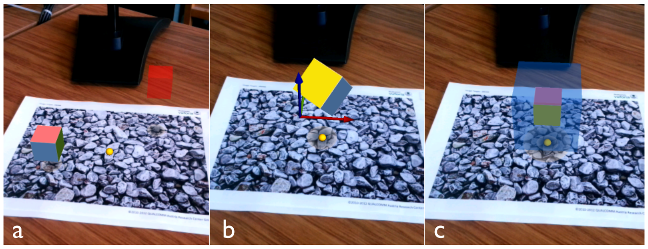

Figure 3: *Three tasks: a. Translation, b. Rotation, c. Scaling.*

4.2 Experimental Procedure

The experiment started with an introduction and participants filling out a pre-experiment questionnaire. Then the participants were given instructions on how to use the prototype system to perform the experimental task. Once they got familiar with the system in the training session (5 minutes for each interface), they proceeded to complete the experimental trials.

The experiment was a within subject design. There were six trials in total with two conditions (using touch and gesture-based interfaces) and three tasks (translation, rotation, and scaling, as shown in Figure 4). The order of the interfaces was counter balanced by alternating them between participants, but the order of tasks was fixed as the purpose of the experiment was not to compare them.

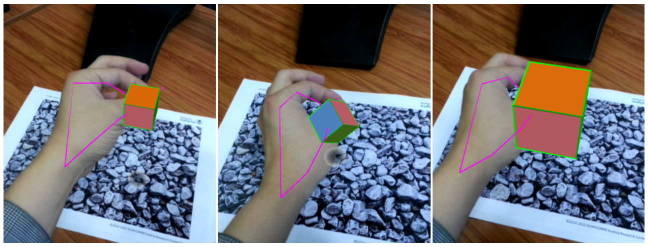

Figure 4: *Examples of gesture-based canonical manipulations for our HAR application by participants.*

In each trial, participants pressed the on-screen start button at the beginning of each task and pressed the stop button when they finished. The task performance was measured in task completion time and placement error, and both were automatically measured and recorded by the system. At the end of each trial, participants answered a usability questionnaire. Once the participant finished all six trials, they were asked to answer a post-experiment questionnaire to give more detailed subjective feedback. The whole user study took approximately 40 minutes on average.

5 Results

We recruited 18 participants (15 males and 3 females) whose ages range from 19 to 32 years old ($M = 24.83$, $SD = 4.25$) for the experiment. All of them except one had previous experience with using a touch-based interface, 12 had used hand gesture-based interface, and 11 had used AR interfaces before. However, their experience of using hand gesture-based interfaces mainly came from the game consoles like Microsoft Kinect[5] and Nintendo Wii[6], and their AR experience were less about interaction but more focused on information display instead. Twelve participants answered right as their

[5] www.xbox.com/kinect

[6] http://www.nintendo.com/wii

dominant hand, while 4 of them answered left, and the other 2 answered both. During the experiment, based on their preference, ten participants held the device in their left hand and used the right hand for interaction, while for eight it was the other way around.

5.1 Task Performance

We measured task completion time and placement error for comparing the user's task performance when using touch and gesture-based interfaces for each experimental task. As both task completion time and placement error in some of the conditions were found not normally distributed (using the Shapiro-Wilk Test), we used non-parametric (Wilcoxon Signed-Rank with α = 0.05) tests for comparing the two conditions.

First we compared the task completion time between the two interfaces (Figure 5). The touch-based interface took significantly less time than the gesture-based interface for the translation task (Z = -2.548, $p = .011$; Touch M = 18.8 sec., SE = 1.71; Gesture M = 26.4 sec., SE = 3.362) and the scaling task (Z = -2.678, $p = .007$; Touch M = 10.7 sec., SE = 0.9; Gesture M = 15.2 sec., SE = 1.2).

Although the gesture input normally just need one spatial movement to complete these two tasks, the user had to move the hand in 3D space over a much larger distance and at a slower speed to keep the hand tracking working. In the touch screen condition, even though the user needed to touch several times with small scrolling motions on the screen to achieve three axis movements, they could do it quickly and over a short distance.

In contrast, the gesture-based interface (M = 25.9 sec., SE = 2.9) took less time than the touch interface on average (M = 30.3 sec., SE = 4.1) for the rotation task, but no significant difference was found (Z = -1.023, $p = .306$).

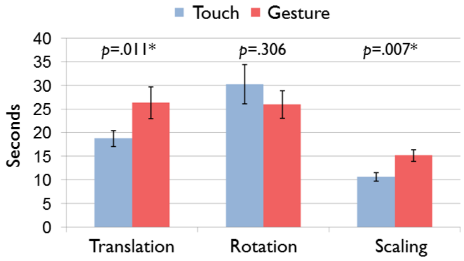

Figure 5: *Task completion time (error bar: SE).*

When comparing the placement error for each task, overall the gesture-based interface had more error on average, but no statistically significant difference was found (translation $p = .215$; rotation $p = .711$; scale $p = .647$; see Figure 6). For the translation task, with the touch-based interface participants made average errors of 6.2 mm (SE = 2.0) while with the gesture-based interface the error was 7.9 mm on average (SE = 2.6). The touch-based interface had 1.4 degrees of error on average (SE = 0.3) in the rotation task, while the gesture-based interface had 5.7 degrees of error on average (SE = 4.5).

One main factor of these difference was the accuracy of the hand pose tracking and estimation software. The average amount of the estimated joint and fingertip location error to the ground truth was around 5 mm, while the hand pose error was much bigger because it was calculated and estimated based on the joint angle. Another possible reason would be shakiness from keeping the hand in the midair.

For the scaling task, touch and gesture-based interfaces had 1.8 and 1.9 mm of errors on average (SE = 0.3 and 0.5), respectively.

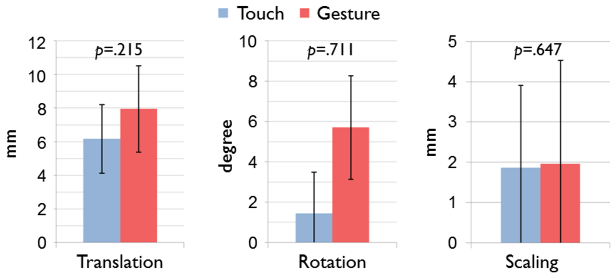

Figure 6: *Placement error (error bar: SE).*

5.2 Usability Questionnaire

After using each interface for each task, participants answered a questionnaire with eight subjective questions (see Table 1) about different aspects of usability of the interface. The questions were answered in seven level Likert-scale rating ranging from 1 (totally disagree) to 7 (totally agree).

We compared participants' answers between the two interfaces (touch and gesture) for each experimental task with Wilcoxon Signed-Rank (α = 0.05).

Table 1: *Usability questions*

	The given interface was:
Q1	Performing well
Q2	Easy to use
Q3	Easy to learn
Q4	Intuitive – you feel straightforward to learn and use the method without much thinking
Q5	Natural – you feel the method similar to your daily interaction behaviors
Q6	Useful for tasks
Q7	Mentally stressful
Q8	Physically stressful

For the translation task, both interfaces were rated above 6 on average for Q1, Q2, Q3 and Q6, reflecting that the participants felt they were performing well, the interface was easy to use, easy to learn, and useful to complete the tasks. Participants rated both of the interfaces having not much stress neither mentally (Q7. Touch(T): Md = 1, Gesture(G): Md = 2) nor physically (Q8. T: Md = 1.5, G: Md = 2.5). No statistically significant difference was found between the two interfaces in all of these six questions. However, for the questions asking how intuitive (Q4) and natural (Q5) was the interface, participants gave significantly higher (Q4. Z = -2.263, $p = .024$; Q5. Z = -3.007, $p = .003$) rating to the gesture-based interface (Q4. Md = 7, Q5. Md = 7) rather than the touch-based interface (Q4. Md = 6, Q5. Md = 5). The scaling task showed similar results as the translation task, except that only the naturalness (Q5) showed a statistically significant difference, which indicated that the gesture-based interface (Md = 6) was rated significantly more intuitive (Z = -2.263, $p = .024$) than the touch-based interface (Md = 5).

In terms of the rotation task, participants gave a higher rating to the gesture interface for the first six questions (Q1$\sim$6), and there was a significant difference for four questions (Q2$\sim$5). These results show that participants felt the gesture-based interface (Q2. Md = 7; Q3. Md = 7) was significantly easier to use (Q2. Z = -2.226, $p = .026$) and learn (Q3. Z = -1.996, $p = .046$) compared to the touch-based interface (Q2. Md = 5; Q3. Md = 6). Meanwhile, the gesture-based interface (Q4. Md = 6.5; Q5. Md = 6) was more intuitive (Q4. Z = -2.994, $p = .003$) and natural (Q5. Z = -3.044, $p = .002$) than the touch-based interface (Q4. Md = 5; Q5. Md = 4). No significant difference was found for Q6 and Q7 as most of the participants felt not much stressful for both interfaces (Md = 2 or less for all).

Comparing the averaged ratings across the all tasks showed that the gesture-based interface was significantly more intuitive (Z = -3.162, $p = .002$) and natural (Z = -3.395, $p = .001$) compared to the touch-based interface, while both of the interfaces were well accepted by the participants with getting rated around 6 or above for performing well, being easy to use and learn, and useful, yet not being much stressful. Figure 7 presents a box plot that summarizes this result. With good internal consistency (Cronbach's α = .858), comparing the average of the ratings across all questions and tasks showed that overall the gesture-based interface (M = 6.2, SE = 0.126) received significantly higher (Z = -2.391, $p = .017$) ratings for the usability questions compared to the touch-based interface (M = 5.8, SE = 0.172).

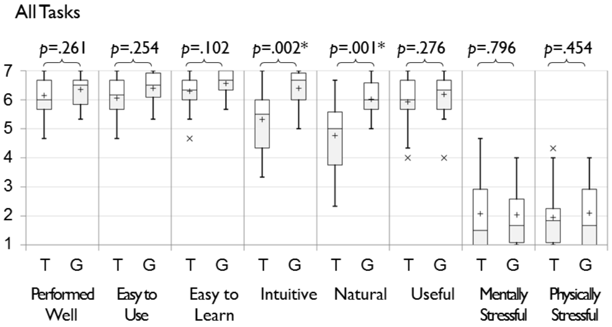

Figure 7: *Results of usability question in Likert-scale with all type of tasks aggregated (T: touch, G: gesture; +: mean, ×: outlier with 1.5 IQR, *: statistically significant).*

5.3 Post-experiment Questionnaire

At the end of the questionnaire, we also asked participants to choose an interface that they prefer to use for each experimental task and in general with three tasks integrated (see Figure 8). Gesture-based interface was preferred by more than half of the participants (10 and 11) for doing the translation and rotation tasks, while half of the participants chose it for the scaling task. Only 4 participants chose the touch-based interface for the translation and scaling tasks, and 6 chose it for the rotation task, while the rest chose to use both. For general task completion, 8 participants preferred the gesture-based interface, while 7 preferred the touch-based interface, and the rest choosing both.

Based on the responses to the open questions that we asked at the end of the study, we can explain the users preference above to some extent. First, people felt that the translation is easier to accomplish with the gesture interface, since the operational object and the input gesture are located in the same 3D coordinate system. This makes gesture manipulation very natural and intuitive. In contrast, the touch-based translation is only easier for 2D movement and not convenient for specifying 3D displacement. For example, in our case, extra two finger touching is required for accessing the third axis for the touch-based interface.

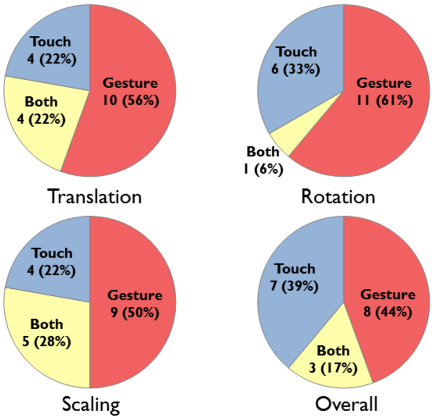

Figure 8: *Users' preference.*

A few users claimed that it could be better for the translation task if both touch and gesture-based interfaces were integrated together and available at the same time, so they can easily switch between these two methods. For example, the gesture control can be used for large scale placement while the screen-touch for adjusting the final position to achieve more precise results at the end. This combination can also be applied when the user needs to complete a time consuming translation with high accuracy requirement. The gesture input might be easily for users to use, but their hands and arms might suffer from fatigue when they have been doing the gesture interaction for a long time, then they can change to use the touch interface instead to make it less physically stressful.

The rotation design also requires double finger touching to control the third axis rotation. Even only using a single finger touch but with two axes activated, which is commonly designed in the mobile applications when people make a 2D touch movement on the screen, the rotation can be very confusing for people. In contrast, rotation based on the hand pose is much more natural and easy to follow, while with less physical stress using a single hand gesture, comparing with the separated multi-touching required for different axes.

Scaling is conducted with similar simple scrolling gesture on the 2D screen and space movement in 3D space for touch and gesture interaction respectively. People felt that the touch scaling was easy to use but not so intuitive for them because they were already used to the two finger pinching for resizing pictures on their personal mobile devices. However, in our case the method is slightly different since we use one finger only.

The overall preference is so much closer between touch and gesture than the other three individual preferences. This is not unexpected since the participants gave high rating to both interfaces in terms of their performance, and they claimed that with both methods they could accomplish tasks efficiently, although each of them has their own advantages and drawbacks.

6 Discussion

Users felt that they need to learn and practice to use the touch-based interface and sometimes instructions are needed because the interface does not mimic the interaction pattern in their real life . Another main shortcoming of touch-based interface is the limited screen size. The finger can be easily cross the screen range with a close AR object view while moving a virtual object further away. Miss touching and shaking problems can also reduce the final performance.

Although the user could feel physical stress by using both hands for holding and interacting with content in the gesture-based interaction methods, they claimed that the gesture input is much more enjoyable to use with less learning and thinking.

Currently the biggest drawback for both input techniques are the depth perception issue. The visual illusion is the main reason that happens to the users frequently because of the viewpoint in 3D AR environment. For example, some users believed that they finished their translation tasks just because both cubes were overlapped with each other from their current perspective, but when they check the result from other viewing angles, they found that the cubes are not aligned at all. It took most people a few seconds to locate the 3D position of the target object even with the help of shadowing and a 3D indicator. Another reason would be lack of the depth occlusion. Since the virtual object is rendered on the top layer of the final AR view, some occlusion from hand that normally happened in the real world sense was not displayed correctly, and this will give more difficulty to users to understand the spatial relationship. The problem can be solved by adding the depth occlusion for the interacting hand.

Depending on the task, both touch and gesture input methods could be chosen by users. Furthermore, being able to use bimanual gesture interaction was expected by users in our study.

7 Conclusion and Future Work

This research investigated the usefulness of natural 3D freehand gesture for one-handed HAR interaction. We present a novel client-server gesture framework that allows us to easily simulate and evaluate different HAR gesture interaction methods. Using this framework we developed a pinch based 3D gesture input method, with which users can manipulate AR objects for translation, rotation and scaling in 3D space.

From the study results, we concluded that the 3D gesture interaction method is as easy to learn and use as the traditional touch-based input. Although it is not very suitable for long period AR manipulation tasks because of the physical stress, it was considered much more intuitive and natural to use, and got a higher rating then the touch-based interface in all user preferences for the canonical 3D HAR interaction.

However, gesture-based interaction needs to be improved in terms of task performance. Improving accuracy of the hand tracking and developing novel gesture-based interaction methods will help to achieve this. Also, using gestures in combination with other modalities of interfaces could be an interesting future research direction.

The implemented interaction technique is just one of many different types that could be tested. There are a number of ways this research could be extended. In the future, we would like to further develop our efforts in designing interaction methods for other AR scenarios. We would also like to investigate the interfaces combining gesture with other interaction modalities, like speech recognition and eye tracking, to evaluate the multimodal interfaces on handheld and wearable AR devices.

References

Bai, H., Gao, L., El-sana, J., and Billinghurst, M. 2013. Work-in-progress: Markerless 3D Gesture-based Interaction for Handheld Augmented Reality Interfaces. In *Proceedings of the 16th IEEE International Symposium on Mixed and Augmented Reality*, IEEE Computer Society, Washington, DC, USA, ISMAR '13, 1–6.

Bai, H., Lee, G. A., and Billinghurst, M. 2014. Using 3D Hand Gestures and Touch Input for Wearable AR Interaction. In *Extended Abstracts on ACM CHI 2014 Conference on Human Factors in Computing Systems*, ACM, New York, NY, USA, CHI EA '14.

Baldauf, M., Zambanini, S., Fröhlich, P., and Reichl, P. 2011. Markerless Visual Fingertip Detection for Natural Mobile Device Interaction. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, ACM, New York, NY, USA, MobileHCI '11, 539–544.

Chen, L.-H., Yu, C.-J., and Hsu, S.-C. 2008. A Remote Chinese Chess Game Using Mobile Phone Augmented Reality. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, ACM, New York, NY, USA, ACE '08, 284–287.

Chun, W. H., and Höllerer, T. 2013. Real-time Hand Interaction for Augmented Reality on Mobile Phones. In *Proceedings of the 18th International Conference on Intelligent User Interfaces*, ACM, New York, NY, USA, IUI '13, 307–314.

Henrysson, A., Marshall, J., and Billinghurst, M. 2007. Experiments in 3D Interaction for Mobile Phone AR. In *Proceedings of the 5th International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia*, ACM, New York, NY, USA, GRAPHITE '07, 187–194.

Hürst, W., and Wezel, C. v. 2013. Gesture-based Interaction via Finger Tracking for Mobile Augmented Reality. *Multimedia Tools and Applications 62*, 1 (January), 233–258.

Kim, H., Reitmayr, G., and Woo, W. 2011. Interactive annotation on mobile phones for real and virtual space registration. In *Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality*, IEEE Computer Society, Washington, DC, USA, ISMAR '11, 265–266.

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*, ACM, New York, NY, USA, UIST '12, 167–176.

Langlotz, T., Mooslechner, S., Zollmann, S., Degendorfer, C., Reitmayr, G., and Schmalstieg, D. 2012. Sketching Up the World: In Situ Authoring for Mobile Augmented Reality. *Personal Ubiquitous Computer 16*, 6, 623–630.

Mossel, A., Venditti, B., and Kaufmann, H. 2013. 3DTouch and HOMER-S: Intuitive Manipulation Techniques for One-Handed Handheld Augmented Reality. In *Proceedings of the 15th International Conference of Virtual Technologies*, ACM, New York, NY, USA, VRIC '13, 12:1–12:10.

Seo, B.-K., Choi, J., Han, J.-H., Park, H., and Park, J.-I. 2008. One-handed Interaction with Augmented Virtual Objects on Mobile Devices. In *Proceedings of the 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, ACM, New York, NY, USA, VRCAI '08, 8:1–8:6.

Wang, R., Paris, S., and Popović, J. 2011. 6D hands: Markerless Hand-tracking for Computer Aided Design. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, ACM, New York, NY, USA, UIST '11, 549–558.