Bimanual Natural User Interaction For 3D Modelling Application Using Stereo Computer Vision

Roy Sirui Yang
Department of Computer Science
University of Auckland
syan095@aucklanduni.ac.nz

Anthony Lau
Department of Computer Science
University of Auckland

Yuk Hin Chan
Department of Computer Science
University of Auckland

Alfonso Gastélum Strozzi
Department of Computer Science
University of Auckland

Christof Lutteroth
Department of Computer Science
University of Auckland

Patrice Delmas
Department of Computer Science
University of Auckland

ABSTRACT

This paper presents a system that allows the user to perform 3D modelling and sculpting using postures and 3D movements of their hands. The system utilises the concept of a Natural User Interface using computer vision techniques. This enables the user to operate 3D modelling software. The system's bimanual control allows left hand postures to select control mode commands, while the right hand controls movements. To evaluate the real world performance of the concept of motion and hand-posture-based control in 3D modelling, a usability test with 10 people was conducted. Participants were asked to perform test tasks that involved moving an object in 3D space. These participants performed the tasks multiple times while being timed, both with the mouse and using the 3D hand tracking system. The results indicated that participants who used the hand tracking system completed the tasks more quickly than those who used the mouse. However, approximately half of the participants reported that they found it easier to use the mouse than the hand tracking system. Overall, the participants reported that they enjoyed using the system.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Representation**]: User Interfaces—*input devices and strategies*; I.5.4 [**Pattern Recognition**]: Applications—*computer vision, face and gesture recognition*

General Terms

Human Factors; Design; Measurement.

Keywords

Natural user interface; Kinetic user interface; bimanual interaction; Computer vision; Real-time 3D hand tracking; Posture recognition; Stereo vision.

CHINZ '12, July 02 - 03 2012, Dunedin, New Zealand Copyright 2012 ACM 978-1-4503-1474-9/12/07 $15.00.

1. INTRODUCTION

With the recent development of digital art and the advancement of powerful personal computers, more and more aspects of life are being augmented or replaced by virtual and digital content. As digital animations are becoming a new form of movies and theatrical plays, digital models are used instead of physical prop. Compared to traditional props, digital models are easier to handle, copy and modify, and they can last longer.

3D modelling software is produced both commercially (Maya [15]), and as open-source (Blender [2]). Digital 3D models may combine basic geometric shapes to form more complex shapes. The user must define precisely the parameters of each shape, and how the shapes are combined. Traditional analogue input devices, such as the computer mouse, can only input movement and direction in 2 dimensions. The keyboard also contains arrow keys that operate in 2 dimensions only. This can cause problems when the user is working with 3D models. The software often facilitates a conversion from 2D coordinates provided by the user into 3D coordinates to be used by the program. To gain mastery of model construction, one needs to learn how a software performs the conversion and control the mouse accordingly. Since conversions of 2D to 3D coordinates do not come naturally, users may be taxed by cognitive overload and are prone to make mistakes.

To address this, our study explores an alternative input method to make 3D model construction easier and more natural for the user. Our system uses computer vision techniques including 3D hand tracking and posture recognition to provide a mouse- and keyboard-free system that allows 3D input. Utilising a stereo computer vision approach, the system is able to track hands in 3D space, and respond to movements accordingly. As there is a high correlation between the user's movements and the actual input received by the system, hypothetically a more natural user interaction should be created [20].

This study aims to explore the usability of 3D input using bimanual control to perform digital modelling and sculpting in 3D space. More specifically, we ask whether hand posture and motion are more natural input methods than the mouse, and if a combination of posture and hand movements are a good way to make use of both hands.

The rest of the paper is organized as follows. The literature review introduces existing technologies and applications related to this project. Section 3 describes the system and the methods used for 3D tracking and posture recognition. Section 4 provides details of a usability study conducted using a prototype of the system. Finally, Section 5 concludes with comments on possible future extensions.

2. LITERATURE REVIEW

Many applications already exist that use computer vision based tracking to realize the Natural User Interface, both as commercial products such as the Wii [18] and the Kinect [17], and as academic research. These applications typically tracks the user's body and respond to user's motion.

The Nintendo Wii is a gaming console which tracks the user's hand through the use of the Wii remote [12] held in the user's hand. The tracking uses multiple infrared beacons at the console side, with a high speed infrared camera on the tip of the remote. By locating and detecting the orientation of the beacon pattern from the console, the Wii remote can calculate its position in 3D space.

The Playstation Move has a very similar mechanism [23]. The user holds a remote controller that emits controllable coloured light unique to the background environment. The system uses the colour and size of the controller's light to place the user's hand in 3D space. Both the Wii and Move also have built in accelerometers in their controllers to increase the accuracy of the tracking. While these system provide good tracking of the hand's position, the user is forced to hold a controller, which can be unnatural and cumbersome.

The Microsoft Kinect tracks the user's whole body without the use of a remote controller [17]. The Kinect has a colour camera, an infrared projector that projects a built in pattern onto the scene, and an infrared camera that measures how the pattern is distorted by different objects in the scene. Because the projector and the camera are separated by a fixed parallax, the depth of objects can be computed by comparing the received pattern with a pattern taken at a predetermined distance [5]. The Kinect also tracks the user's body and limbs using a skeleton tracking algorithm [22].

It should be noted that these motion based consoles promote movements of the user's whole body. This increases energy expenditure significantly [7, 6], and may cause the user to become tired more quickly, the worst case scenario being muscle and joint fatigue and injury [19].

Microsoft is currently working on a product called Holodesk [16]. The system uses the Kinect to detect any objects within the work bench, and transform them into particle clouds in 3D space. This allows the user to interact with virtual objects, and to collaborate with other Holodesk users. For virtual feedback, the system uses a webcam to track the user's head, and generates 3D holographic like visual feedback to the user. This creates a more immersive interaction experience for the user.

Laundry et al. [11] proposed a system that allows the user to interact with 3D objects, and create 3D sketches using their hands. The system uses two infrared cameras (Nintendo Wii remote) to track an infrared emitter attached to the user's finger to achieve 3D hand tracking. This paper explored a number of method that can be used for command selection, including holding up signs, select commands using the foot, and using a clicker pen. The usability study shows that the participants enjoyed using the system, especially creating 3D sketches. The paper concluded that clicker pen method was the most preferred.

Kurata [10, 9] et al. proposed a system called the Hand Mouse. This system consists of a head mounted camera, a clip on display on glasses for visual feedback and a laptop. The system analyses the position of the user's hand, and the object the user is pointing at. It gives feedback to the user if the object pointed at has additional information. While this system conveniently allows the user to interact with the environment, the gear the user has to wear weighs 2.4 Kg, consisting of a head set, glasses, an ear set and a watch, and requires a laptop nearby. This set up is difficult to carry around, and the weight of the gear reduces its usability.

Levesque et al. [13] describes a system that uses both hand to control and interact with visual displays. The system uses data gloves to extract information about the position and posture of the hands. The left hand is used to select commands while the right hand is used for movement control. The system that the paper presents uses a complex system of hand posture. However some of the hand postures have no relationship to their corresponding commands. This means that the user has to memorise all postures, and corresponding commands to make use of the system. This may reduce the usability of the system.

Schlattman et al. [21] developed a system that allows three low cost cameras to track both hands in real time. This system uses background subtraction to segment the hands in all three cameras, then calculates the 3D convex hull of both hands. By locating the position of the finger tips, hand postures can also be recognised. The paper proposed two applications for the system. The first application was a flight simulator, operated using both hands. The left hand was used to select commands via postures, while the right hand was used to move and rotate views. The second application was a mesh editing interface, which allowed the user to select groups of vertices in 3D space. Again, the left hand was used to select command via hand postures while the right hand moved a 3D cursor around.

A system developed by Wang et al. [25] tracked a hand's position in 3D and posture using a coloured glove. The user wears a glove, with different regions having distinctive colours. The paper proposed several applications including digital animation creation, object manipulation and communication using sign language. This system used a single camera, controlled by a single hand wearing gloves, and focused more on posture analysis.

As can be seen, much research has been done relating to motion based applications. While some research uses alternative command selection methods, such as a clicker pen, signs, or pointing with a fingertip, our system used the hand posture approach. Our system aimed to create a low cost system that is affordable to ordinary users, hence we do not use more expensive equipment such as data glove, or retina displays. Our system also used markerless tracking, which does not have any extra requirements for the user. This created a more natural user experience.

3. REAL TIME 3D HAND TRACKING AND POSTURE RECOGNITION

Two low cost camera systems (Micrsoft Xbox Kinect and Minoru Stereo Webcam) available to our lab were compared with two other more expensive stereo camera systems (Custom build uEye stereo

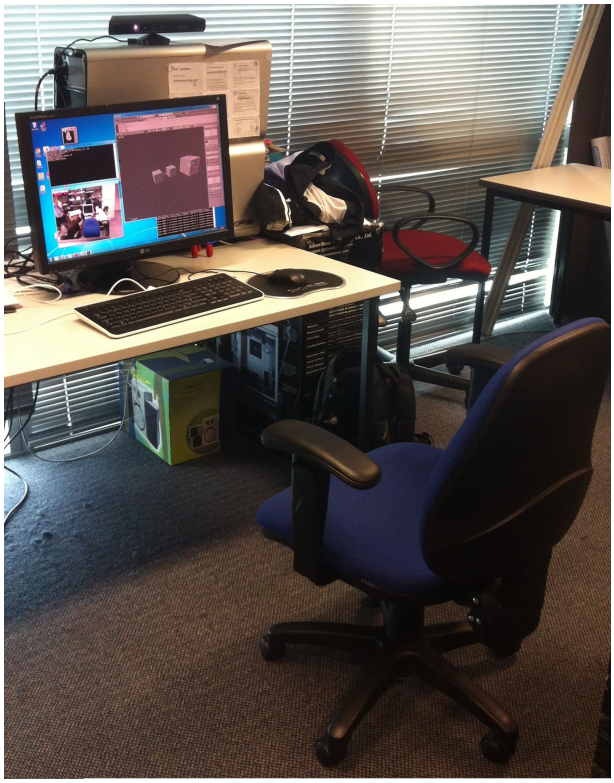

Figure 1: The physical set up of the system. The user is seated in a chair with arm rests in front of the monitor screen. The Kinect sensor is positioned slightly behind and above the screen.

system and FujiFilm W1 camera). We investigated both the intrinsic and extrinsic properties of the systems, as well as their respective costs (Table 1).

The Kinect had the smallest focal length, and a slightly higher distortion value compared with the Minoru. The depth resolution (Z_{res}), or the maximum depth measuring ability was also calculated and compared (Figure 2). The depth resolution measures the real world distance corresponding to 1 pixel disparity at different distances. This, therefore, is the minimal distance detectable by a stereo camera system at a given distance.

The Microsoft Kinect sensor was used in this study. This sensor has a much wider field of view (FoV) than the Minoru, and a much

Table 1: Comparison of four types of stereo camera systems: the Kinect sensor, the Minoru stereo webcam, the uEye cameras and the FujiFilm W1 stereo camera

camera	Minoru	Kinect	uEye	W1
price (USD)	89	150	3000 × 2	600
f	874	513.2	770.5	4372
Distortion κ_1	-0.1378	0.2695	-0.347	-0.087
speed (fps)	30	30	87	30
FoV(H) °	41.9	63.2	52.0	45.3
FoV(V) °	32.1	50.0	34.6	34.5
Z_{res}@1500 mm	44.4	7	adjustable	6.63

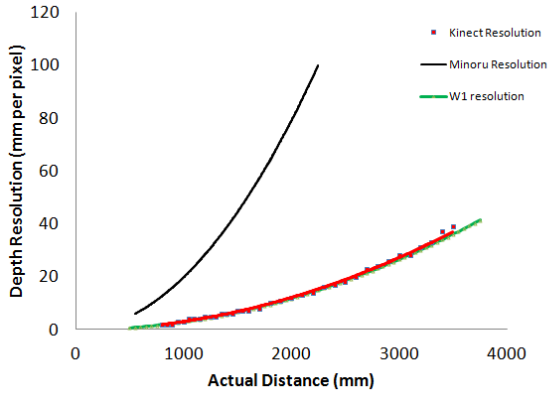

Figure 2: The depth resolution of three stereo camera systems is compared. The depth resolution measures the minimum change in depth that can be detected by a stereo camera system at given distances.

better ability to distinguish depth (Depth resolution), similar to that of the much more expensive W1 camera. The Kinect sensor also provides video streams of depth and colour in real time. This removes the computationally intensive task of depth computation. From the two streams provided by the Kinect sensor, the 3D locations of the hands and the posture of the left hand are extracted. Our system uses a bimanual mode of control - the left hand of the user manages the command selection via hand postures, while the right hand defines direction and magnitude of movement. Our system can easily be customised to accommodate left handed users, interchanging the functions of the left and right hand.

Due to the minimum range of the Kinect sensor, the user is positioned approximately 1400 mm from the camera, and directly in front of a computer screen to provide visual feedback. The camera is placed some distance behind and slightly above the monitor so that the monitor does not obstruct the view of the camera. This design allows the user to be inside the depth sensing range while not being too far away from the computer screen (Figure 1).

The user's seat is provided with arm rests. This allows the user's left hand to be positioned comfortably on the arm rest, significantly reducing strain. The right arm, which performs larger movements while suspended in the air, is likely to experience fatigue when using the system.

3.1 The Kinect Sensor and its Limitations

Our system uses the Microsoft Kinect sensor. The system consists of a colour CMOS camera, an infrared projector, and an infrared CMOS camera. The system provides real time video stream at 30 frames per second, as well as real time depth stream also at 30 frames per second. The high frame rate means that if the processing is fast enough, the system can be used for real time interaction. Both streams operate at a resolution of 640×480. After calibration using Zhang's method [27] we determined that the colour camera has a focal length of 513.2, a horizontal field of view of $63.43°$, and a vertical field of view of $50.00°$. These angles limit the amount of space the user can move in, and potentially lead to interruptions during the interaction when the user leaves the field of view.

The infrared camera is placed next to the colour camera. Due to the

Figure 3: The depth image (grey scale) is overlaid with: (left) the original colour image and (right) the corrected colour image.

difference in both positions, orientation and intrinsic parameters (focal length, principal points, distortion etc.), the objects in the colour image have different positions in the depth image (Figure 3).

By using a function provided by the Microsoft Kinect SDK, we are able to correct the misalignment by mapping the colour image onto the depth image. The detailed algorithm of the function is purposefully undisclosed, hence we do not know exactly how the mapping was done. After measuring 15 sample points, we found that the colour image and the depth image had a mean alignment error of 19.4 pixels (distance of the same object in depth and colour images), while the mean error for corrected colour images was 6.2 pixels. A paired sample t-test showed that the corrected images had a statistically significantly smaller error ($t = 6.8, p < 0.0001$). To improve the tracking accuracy, the colour images are *always* corrected. The colour video stream is calibrated after the correction, and the focal length is 588.9, with the radial distortion coefficients: ($\kappa_1 = 0.2695, \kappa_2 = -0.8262$). The error correction and the calibration parameters are important to accurately determine the hand position and ensure that there is a high correspondence between a user's movements and the system's response.

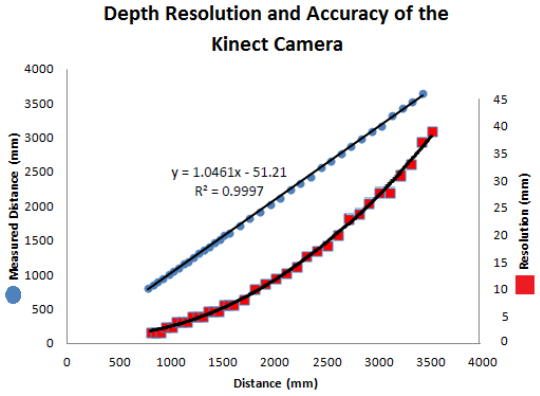

Figure 4: Measured depth accuracy and resolution of the Kinect at different distances.

Due to the geometry of Stereo Vision, the resolution of the depth values decreases as objects get further away from the cameras (Fig. 4). This means that the user should be as close to the Kinect as possible for optimal accuracy. To prevent the user's hand moving too close to the Kinect (which has a minimum measurement distance of 800 mm), we recommend using the system at a distance of 1400 mm.

3.2 3D Hand Tracking

The Continuous Adaptive Mean Shift algorithm (CAMShift) [3] is used to track the user's hand based on the colour of the skin. A 2D histogram of hue and saturation is first constructed by sampling the user's skin. By treating the histogram as a probability distribution of a skin pixel, the colour images are converted into a probability map. Since the system tracks the hand based on skin colour, problems often arise when objects of similar colour are present (such as the face). We address this issue by tracking three blobs of skin independently (the left hand, right hand and the face)(Fig. 5). Human skin can sometimes be very bright due to its reflective properties. Hue (H) can be unstable at extremely dark and bright pixels. To address this problem, the system treats all pixels that have a total brightness $100 < (R + G + B) < 740$ (where R, G, B are the red green and blue channel of the pixel in the RGB colour space) as having a probability of 0. CAMShift is then used on the probability map to track pixel blobs that have a high probability of being the skin. The depths of the hands are extracted from the aligned depth map provided by the Kinect.

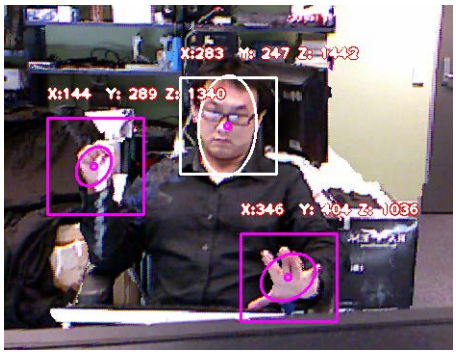

Figure 5: An example of the stereo tracking unit at work: 3 blobs of skins are being tracked, and marked by rectangles (search windows for CAMShift) and ellipsoids that mark the width, length and rotation of the tracked objects.

The hard part of 3D hand tracking is the acquisition of the 2D image and depth coordinates. Once this is accomplished, the 2D X and Y coordinates can be easily translated into real world coordinates using the depth information and the equation:

$$X_w = \frac{X \times Z_w}{f} \qquad Y_w = \frac{Y \times Z_w}{f} \tag{1}$$

X_w and Y_w are the real world X and Y coordinates of a pixel, X and Y are the corresponding coordinates in the image, and f is the focal length of the camera. The final 3D location of the hand is obtained after being smoothed using a Kalman filter.

3.3 Posture Recognition

3D tracking of the hand removes the need for a mouse, as all *movement* is controlled by the user's right hand. However, simple movement control is not sufficient to fully operate a 3D modelling system. To facilitate a natural user interaction for command selection, we used vision based posture recognition. The hands were first segmented using a depth filter:

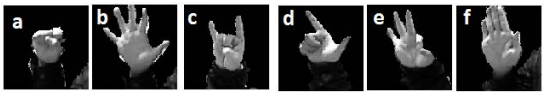

Figure 6: Six postures currently in our database. They correspond to the commands: *a) Move Object, b) End/Pause Command, c) Reset Scene, d) Rotate View, e) Reserved, f) Translate View*.

$$P_h(X,Y) = \begin{cases} \text{true} & \text{if } |D(X,Y) - d_h| < 100 \text{ mm} \\ \text{false} & \text{if } |D(X,Y) - d_h| \geq 100 \text{ mm} \end{cases} \quad (2)$$

where $P_h(X,Y)$ is a boolean variable representing if the pixel at (X,Y) belongs to the hand; $D(X,Y)$ is the depth value in mm; d_h is the current depth value of the hand, obtained from the 3D tracker. The threshold of 100 mm is chosen, as studies have shown that the average length of a human hand is about 172.2 to 189.0 mm [1].

By using the following equation (derived from using similar triangles), the projected size of the hand is determined:

$$S_w = f \times \frac{S_h}{d_h} \quad (3)$$

S_w denotes the window size of the segmented hand (pixels) on the colour image, f denotes the focal length of the camera, S_h denotes the real world size of the hand. A value of 250mm is chosen to accommodate variation in the population. d_h denotes the depth of the hand. After the hands have been segmented the image is scaled to a 100×100 image. The images extracted are the same size regardless of position and distances. This allows more robust posture recognition.

The recognition uses Principle Component Analysis (PCA). PCA can be used for recognition tasks such as posture recognition [8, 4], face recognition [24] etc.. A database of six hand postures (14 training images each) was first constructed to train the system. By treating the images as vectors, PCA projects these vectors onto eigenvectors calculated from the database. By using the K-Nearest Neighbour method, the closest matches of known postures from the database are found.

With the postures in Figure 6, the user can perform basic 3D modelling operations in Blender. The system is tested on the first author and 3 other people. Recognition rates of the hand postures are shown in Tables 2 and 3. It should be noted that the database was constructed using images of the author's hand. Hence the actual recognition rate is likely to be lower with other people (Table 3).

When the recognition rate is low, the posture is very difficult to use due to mismatches. Therefore, in practice, only postures *a,b,c* and *e* can be reliably used.

After posture recognition is completed, the coordinates and the code for the hand posture is sent to the 3D modelling software Blender [26] via a TCP connection. By using internal Blender scripting, the user's input is translated into actual 3D modelling operations, and visual feedback is provided to the user.

Currently, our system supports several modelling functions, each having their corresponding hand postures: a) Move Object, c) Scene Reset, d) View Rotation, f) View Translation. Command selection can also be done using the keyboard for testing purposes. Users can begin a command by performing its respective posture (a, c, d, f), and pause using posture (b).

Table 2: Confusion Matrix of the posture registered in the database (%)

test \ match	a	b	c	d	e	f
a	100	0	0	0	0	0
b	0	100	0	0	0	0
c	3.3	0	96.7	0	0	0
d	13.3	6.7	0	80	0	0
e	0	0	13.3	16.7	70	0
f	0	6.7	0	0	3.3	90

Table 3: Posture Recognition Rate for Participants not Included in the Database

Posture	Recognition Rate (%)
a	100
b	100
c	89
d	67
e	89
f	56

To move an object, the user simply needs to perform posture (a), and the object will move in 3D space corresponding to the movement of the user's right hand. View rotation is done by using posture (d), and the right hand movement is used analogously to rotation as done with the mouse, using a track ball approach. The view translation is similar to the move object command, except that posture (f) is used and the camera position is moved instead. When the left hand is performing posture (b), the system will ignore any right hand movements. In some cases, the user needs to move their right hand across a large distance. This can result in awkward body posture, or the hand going out of the camera's view. In these cases, the user is able to use the pause command in the middle of the movement, adjusting the right hand to a more comfortable position, and continuing the movement.

4. USABILITY STUDY

4.1 Design and Methodology

We conducted a usability study to test the overall performance of the system. The study had two aims. The first aim was to test the overall performance of the system, both objectively and subjectively. The second aim was to test the usability of the HCI design of the system. This includes how well users like the overall design of the bimanual method and how comfortable the postures and movements are when controlling the system.

4.1.1 Experimental Procedure

The participants were required to complete tasks using both the mouse and the 3D tracking system under experimental conditions while being timed. The tasks involved moving an object in 3D space to a specific position (Figure 7). A total of eight tasks were prepared. For each task, the initial position of the movable cube was different, but the final position was the same for all tasks and conditions.

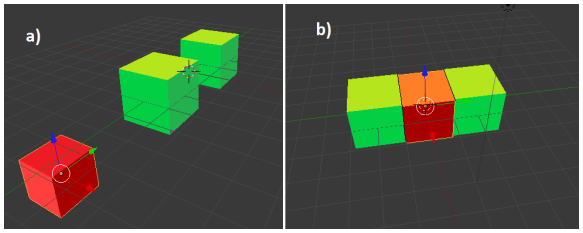

Figure 7: An example of the task used in the study. The red cube at an initial position (a) must be moved between the green cube pair (b). The eight tasks use a previously randomly generated 3D location as the initial position of the red cube, and the green cubes (final position) will be the same for all tasks.

To take account of the order effect, the participants were separated into two groups. Group A performed the eight tasks using the mouse (mouse condition) first followed by the 3D hand tracker (hand condition), and vice versa for group B. For each condition, the participants were introduced to the input device, and taught how to perform the tasks. The participants then had some time to practice until they were comfortable with the condition. When they were ready, they completed the eight tasks, using the input devices specific to their group. They were then given a short break while the system was set up for the next condition. The above steps were repeated for the second input devices, including completion of the eight tasks, again using a different input method.

At the end of the two sets of tasks, a questionnaire was administered to the users asking their opinions on the overall system. The questionnaire required the users to rate, on a 7 point Likert Scale [14], the extent to which they agreed (1) or disagreed (7) with the questions.

4.1.2 Independent and Dependent Variables

The independent variable was the input method of the system. This had two conditions - the (M) mouse and the (H) 3D hand tracking. This study compares the usability of these two input devices. Group A participants had an order of condition (M) followed by (H), while group B participants had condition (H) followed by (M).

The dependent variables were the time taken by the user to complete the task (objective measurement), as well as a questionnaire that measures user's opinion on the system (subjective). The questionnaire also asked the user, in the form of open ended questions, what he/she liked or disliked about the system, and his/her opinion of the bimanual framework of control.

4.2 Results and Discussion

10 participants joined this study. Participants 1,3,5,7,9 formed group A. Participants 2,4,6,8,10 formed group B. The average age was 26 years. The majority of participants were students from various faculties of the University of Auckland. All participants were computer literate, and competent with the mouse. All participants successfully completed all the tasks.

4.2.1 Quantitative Measurements

The mean time for using the mouse to complete a single task was 14.9 second (std. dev. 3.5 s). The mean time for using the hand tracker to complete a single task was 8.6 second (std. dev. 2.4 s). A paired sample t-test was conducted to compare the average time taken to complete a single task under each condition. The t-test showed that participants using the hand tracker took a significantly shorter time to complete the tasks than the participants who used the mouse ($t = 5.16, df = 7, p = 0.0013$).

Table 4: Average Time for Completing the Tasks

Tasks	Mouse	Hand
1	18.01	8.5
2	13.68	**5.18
3	16.29	**6.5
4	11.34	8.75
5	21.3	12.47
6	14.33	*7.83
7	12.33	8.1
8	11.6	11.2

*$p<0.05$
**$p<0.01$

Table 5: Quantitative result of the questionnaire

Question	Average rating
Easier to use than the mouse	4.0
Tracking system is natural to use	5.5
Posture is natural to use	4.7
Bimanual is natural to use	5.3
Tiring to use the system	5.1
Overall enjoyed using the system	5.4

As participants became more competent with practice, they generally completed the tasks more quickly. Thus, it may be expected that with practice, the time that they take to complete the tasks will further improve. This would apply particularly to the hand tracking system, as computer users are generally familiar with using the mouse. The first author, being a skilled user, is able to completed these tasks using the 3D hand tracker with a mean time of 3.60 second (std. dev. 1.49 s).

Table 5 shows participants' average responses to the questionnaire. Participants generally found the system intuitive to use. Even though the question regarding the ease of use compared to the mouse had a neutral average rating, closer examination of individual responses showed large variations in ratings across all participants.

4.2.2 Qualitative Measurements

Participants' responses to the open ended questions indicated that they liked the hand tracking system because of the natural, intuitive in which it can be used. They also liked the ease of use of 3D input and found it easier to learn and use. This supports with our hypothesis, that 3D input allows movements in 3D with a single motion, instead of multiple movements with rotation of views in between (as how movements in 3D can be performed in Blender).

Many aspects of the system can still be improved. Losing track of the hand can caused frustration during use. This is often caused by hands going out of the screen (field of view of the camera), or hands being too close to the face. To address this issue, the sensitivity of the control could be readjusted, so the magnitude of movement required by the user is reduced. Stability and auto recovery of tracking also need to be further improved.

Participants reported that their right arm (used for 3D movement

control) became tired quickly. This may be caused by the fact that the hands and arms need to be suspended in the air when using the system. As the chair has arm rests, it is possible to adjust the sensitivity of control so that the movement required by the right hand is reduced. This should allow the user to use the arm rest to support his/her right arm at all times. Consequently the arm would not need to be suspend in the air and fatigue would be minimised.

Majority of participant reported when using the bimanual controls, they would have preferred the movements and postures to be combined into a single hand. In addition, they reported that if both hands support posture and movement tracking, a multi-hand support similar to that of multi-touch technology, would allow more flexibility in controlling the system. Future research could investigate the usability of movement and posture combined into a single hand, as well as the multi-hand support.

5. CONCLUSION AND FUTURE WORK

In this paper, we presented a prototype system that uses 3D hand tracking and posture recognition to control a 3D modelling software. The system uses the Kinect sensor to perform real-time 3D tracking. The position of the user's left hand is then extracted using the depth map provided by the Kinect, and PCA is used to perform hand posture recognition. The system uses a bimanual interaction style, whereby the left hand is used to command selection via hand postures, while the right hand is used to control movements in virtual 3D space, using the real world 3D coordinates of the hand. The 3D coordinates of the hands and posture are sent to the open-source 3D modelling software Blender via a TCP connection. By using internal scripting, the user is able to access modelling and sculpting functions inside Blender.

We also conducted a usability study to compared the 3D hand tracking system to the computer mouse when performing 3D object translation. The results showed that participants moved objects in a significantly shorter time when using 3D hand tracking than when using the mouse. The participants found the 3D hand tracking system to be a natural and enjoyable way to perform 3D modelling operations. However, all participants found hand tracking to be tiring over long periods of use.

Overall, the hand tracking approach to modelling appears to be promising and worthy of further investigation. However, there are a number of challenges that need to be addressed. To reduce fatigue induced from using the system, the possibility of better arm support and the reduction of the magnitude of movements necessary to use the system requires further investigation. Furthermore, the current use of postures, which suffers from low recognition rates would benefit from further investigation. We may also explore the use of mixed approaches, including the use of a keyboard, and the combination of posture and movements of a single hand.

6. REFERENCES

[1] A. K. Agnihitri, B. Purwar, N. Jeebun, and S. Agnihotri. Determination of sex by hand dimensions. *The Internet Journal of Forensic Science*, 1(2), 2006.

[2] Blender.org. Blender official site, retrieved: Mar 2012. http: www.Blender.org.

[3] G. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Tech. Journal*, 1998.

[4] N. H. Dardas and E. M. Petriu. Hand gesture detection and recognition using principal component analysis. In *Computational Intelligence for Measurement Systems and Applications (CIMSA), 2011 IEEE International Conference on*, pages 1–6, 2011.

[5] B. Freedman, A. Shpunt, M. Machline, and Y. Arieli. Depth mapping using projected patterns, 2007. Patent, US0240502A1.

[6] L. Graves, N. Ridgers, and G. Stratton. The contribution of upper limb and total body movement to adolescentsâĂŹ energy expenditure whilst playing nintendo wii. *European Journal of Applied Physiology*, 104(4):617–623, 2008.

[7] L. Graves, G. Stratton, N. D. Ridgers, and N. T. Cable. Energy expenditure in adolescents playing new generation computer games. *British Journal of Sports Medicine*, 42(7):592–594, 2008.

[8] D. Jiangwen and H. T. Tsui. A PCA/MDA scheme for hand posture recognition. In *Proc. Automatic Face and Gesture Recognition Conference*, pages 294–299, 2002.

[9] T. Kurata, T. Okuma, M. Kourogi, T. Kato, and K. Sakaue. Vizwear: Toward human-centered interaction through wearable vision and visualization advances in multimedia information processing âĂŤ pcm 2001. In H.-Y. Shum, M. Liao, and S.-F. Chang, editors, *PCM*, volume 2195 of *Lecture Notes in Computer Science*, pages 40–47. Springer Berlin / Heidelberg, 2001.

[10] T. Kurata, T. Okuma, M. Kourogi, and K. Sakaue. The hand mouse: Gmm hand-color classification and mean shift tracking. In *Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, 2001. Proceedings. IEEE ICCV Workshop on*, pages 119–124, 2001.

[11] B. Laundry, M. Masoodian, and B. Rogers. Interaction with 3D models on large displays using 3D input techniques. In *Proc. of CHINZ'10 conference*, pages 49–56, 2010.

[12] J. C. Lee. Hacking the nintendo wii remote. *Pervasive Computing, IEEE*, 7(3):39–45, 2008.

[13] J. C. Levesque, D. Laurendeau, and M. Mokhtari. Bimanual gestural interface for virtual environments. In *Virtual Reality Conference*, pages 223–224, 2011.

[14] R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932.

[15] Maya. Maya official site, retrieved: Mar 2012. http://usa.autodesk.com/maya/.

[16] Microsoft. Holodesk microsoft research, retrieved: Feb 2012. http://research.microsoft.com/apps/video/default.aspx?id=154571.

[17] Microsoft. Official site for microsoft xbox kinect, retrieved: Mar 2012. http://www.xbox.com/en-US/kinect.

[18] Nintendo. Nintendo wii official site, retreived: Mar 2012. http://www.nintendo.com/wii.

[19] Nintnedo. Wii - health and safety precautions, retrived feb 2012. http://www.nintendo.com/consumer/wiisafety.jsp.

[20] A. Parkes, I. Poupyrev, and H. Ishii. Designing kinetic interactions for organic user interfaces. *Special Issue of Communications of the ACM*, 51(6):58–65, 2008.

[21] M. Schlattman and R. Klein. Simultaneous 4 gestures 6 dof real-time two-hand tracking without any markers. In *Proceedings of the 2007 ACM symposium on Virtual reality software and technology*, VRST '07, pages 39–42, New York, NY, USA, 2007. ACM.

[22] A. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real time human pose recognition in parts from a single depth image. *Computer Vision and Pattern Recognition 2011*, pages 1297 – 1304, June 2011 2011.

[23] Sony. Ps3 move faq, retrieved: Feb 2012. http://blog.us.playstation.com/2010/09/07/playstation-move-the-ultimate-faq/.

[24] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[25] R. Y. Wang and J. Popović. Real-time hand-tracking with a color glove. *ACM Trans. Graph.*, 28(3):63:1–63:8, July 2009.

[26] R. S. Yang, A. Lau, Y. H. Chan, A. G. Strozzi, P. Delmas, and C. Lutteroth. Real time 3D hand tracking for 3D modelling applications. In *Proc. of the* 26^{th} *IVCNZ Conference*, 2011.

[27] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.