PinNPivot: Object Manipulation using Pins in Immersive Virtual Environments

P. Christopher Gloumeau, Wolfgang Stuerzlinger, and JungHyun Han

Abstract—Object manipulation techniques in immersive virtual environments are either inaccurate or slow. We present a novel technique, PinNPivot, where pins are used to constrain 1DOF/2DOF/3DOF rotations. It also supports 6DOF manipulation and 3DOF translation. A comparison with three existing techniques shows that PinNPivot is significantly more accurate and faster.

Index Terms—Artificial, augmented, and virtual realities, User Interfaces, Interaction styles, Human-computer interaction

1 Introduction

Object manipulation and placement in virtual environments has been an active research topic for the past two decades. Many different techniques have been proposed based on a variety of input devices. The research has evolved from using keyboards and mice, to touchscreens, and more recently, to techniques that rely on virtual reality (VR) controllers and depth sensors.

Despite this evolution, techniques to manipulate objects in immersive virtual environments (IVEs) either are not able to, or take an unreasonable amount of time to, achieve the level of accuracy required for professional applications such as virtual 3D modeling for product design. Thus, there is a need for innovative object manipulation techniques that are both accurate and fast.

In this paper, we propose a new technique, PinNPivot, where the user can either manipulate the object directly or use *pins*, which can be placed anywhere on the object, to constrain its rotation. To evaluate our technique, we compared it against three representative methods from previous work. The results show that PinNPivot is superior to the other methods in both accuracy and speed.

2 Related Work

Wang et al. [1] introduced a direct six degrees of freedom (6DOF) technique to manipulate objects, where hand motion was directly mapped to object motion. This 6DOF technique is the most intuitive as it mimics the way people manipulate objects in the real world. Unfortunately, due to the limits of how steady a hand can be held in space, accuracy suffers with this technique. Other direct mapping techniques include the work of Bowman and Hodges [2], Pouprev et al. [3], Pierce et al. [4], Araújo et al. [5] and Kim and Park [6].

The PRISM technique [7] aimed to increase accuracy for 6DOF manipulation by allowing the translation distance and rotation angle to be scaled down before being applied to objects. Whereas scaled translation increased accuracy, scaled rotation was perceived as confusing and hindered accuracy [8]. On the other hand, scaling was used with great success to manipulate objects at a distance [9] and to change the user's current view [10].

Bossavit et al. [11] stressed the importance of using appropriate metaphors, as people learn a technique more quickly if it bears some resemblance to known actions or operations. One simple metaphor is to use an object's primary axes. Mendes et al. [12] proposed a separated DOF (SDOF) technique, where the axes have spheres at each end with which the user can translate and rotate the object. This technique significantly increased accuracy compared to direct 6DOF and PRISM, but required more time since every manipulation is only along a single axis [8].

The handle bar metaphor [13] skewers objects with a bimanual handle bar and performs all transformations through that handle bar. They also proposed a crank mechanism for rotation around an arbitrary axis. Bossavit et al. [11] and Cho and Wartell [14] presented similar rotation methods. Caputo et al. [15] proposed to use a bar that extends outward from the center of the object and contains two handles. The user grabs the object for translation, the closer handle for rotation and the farther handle for scaling.

The 7-handle technique [16] allows the user to place seven handles around an object. The handles are split into three levels such that each level offers different manipulation capabilities. For example, the first level handles offer rotations around an arbitrary point or axis when locked.

Mendes et al. [17] proposed mid-air objects on rails (MAiOR). While translating an object in 3DOF, an axis is displayed which can be locked. If locked, the axis acts like a rail and the object like a train, i.e., the object can translate in 1DOF along the axis. Similarly, a circular rail can be locked during 3DOF rotation to restrict the rotation to 1DOF. Based on the results for PRISM, the authors provided scaling for only translation. Yet, their rotation method was perceived as confusing and ultimately reduced accuracy.

3 PinNPivot

We propose *PinNPivot*, a manipulation metaphor that provides users with auxiliary objects, called *pins*, to constrain

- *Christopher Gloumeau and JunHyun Han are with Korea University. E-mails: PCTGloumeau@gmail.com, jhan@Korea.ac.kr.*
- *Wolfgang Stuerzlinger is with Simon Frazer University. E-mail: w.s@sfu.ca.*

Manuscript received Month Day, Year; revised Month Day, Year.

Fig. 1: The three gestures used in PinNPivot.

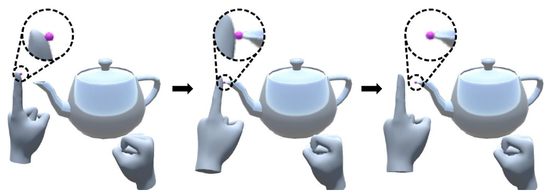


Fig. 2: A pin is created with the `point` and `grab` gestures and becomes stuck to the object surface.

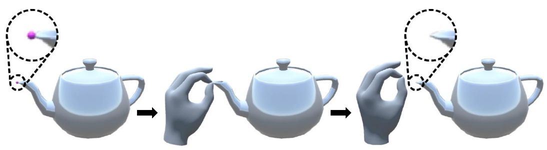


Fig. 3: A pin is deleted with the `pinch` gesture.

Fig. 4: A pin is locked by touching it with the `point` gesture.

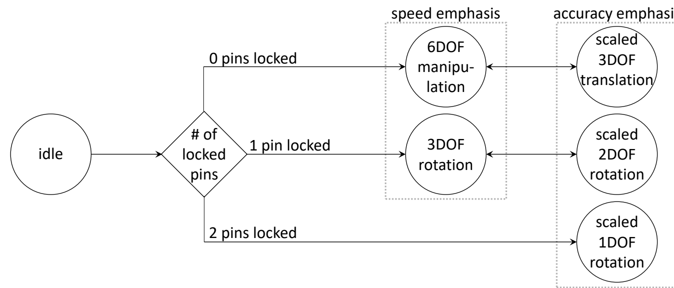


Fig. 5: PinNPivot's manipulation flowchart.

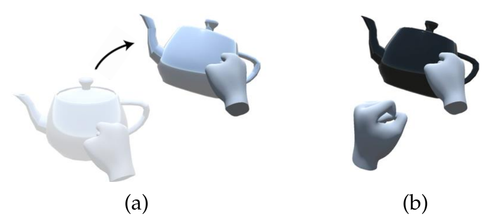


Fig. 6: Manipulation with no pin: (a) 6DOF manipulation. (b) 3DOF translation

the rotation of the object in focus. It mimics pinning a note to a notice board, where the pinned part of the note is stuck to the board but the note can still be rotated around the pin. The pin acts as a *pivot* for rotation.

The benefits of PinNPivot can be understood when compared to a simple 6DOF technique. One of the biggest complaints with direct 6DOF is that, even when a part of the object is accurately placed, it is often unwantedly displaced when another part of the object is moved [8]. In contrast, PinNPivot allows the user to lock an accurately placed part and continue interacting with the rest of the object without having to be concerned about displacing the locked part.

3.1 Pin Creation, Deletion and Locking/Unlocking

In this study, we used the Oculus Rift headset and controllers (which are mapped to virtual hands). PinNPivot relies on three gestures `grab`, `point` and `pinch` (Fig. 1), which are formed by squeezing the trigger buttons on the controllers. PinNPivot is a two-handed technique, where users may utilize either hand at their convenience, as is the case in the real world.

To create a pin, the user must form the `point` gesture with one hand and the `grab` gesture with the other, as shown on the left of Fig. 2. A pin appears at the tip of the pointing finger. When the user touches the object with the pin, it is placed on the object's surface (Fig. 2 center) and remains stuck to the surface (Fig. 2 right).

Deleting a pin is like "plucking" it from the object. See Fig. 3. The user simply has to form the `pinch` gesture over the pin and move the hand away from the object.

To use a pin for object manipulation, it first needs to be *locked*. The user can lock a pin by touching it with the `point` gesture. See Fig. 4. When successfully locked, it changes color from magenta to yellow. To unlock a pin, the user touches the pin again with the `point` gesture.

3.2 Object Manipulation

PinNPivot supports five types of object manipulation, as shown in Fig. 5, where the flow depends on the number of locked pins.

3.2.1 6DOF Manipulation and 3DOF Translation

If no pins are locked, the user can manipulate the object in 6DOF (the first branch in Fig. 5). The user can initiate 6DOF manipulation by touching the object with the `grab` gesture. Then, as they move their hand, the object follows the motion (Fig. 6-(a)). The center of rotation is the hand.

While performing 6DOF manipulation, if the other hand forms the `grab` gesture (Fig. 6-(b)), 6DOF manipulation is switched to 3DOF translation. To indicate that the user has entered the 3DOF translation mode, the object's color changes to black. If the second hand releases the `grab` gesture, 6DOF manipulation resumes.

As indicated by the dotted boxes in Fig. 5, the main purpose of 6DOF manipulation is *speed* whereas that of 3DOF translation is *accuracy*. The general usage guideline is to first use 6DOF manipulation to quickly approximate an object's pose and then use 3DOF translation to accurately place one part of the object. For this purpose, 3DOF translation is *scaled*, i.e., the distance traversed by the hand is multiplied by a scaling factor before being applied to the object. In the current implementation, the scaling factor is $^{1}/_{4}$.

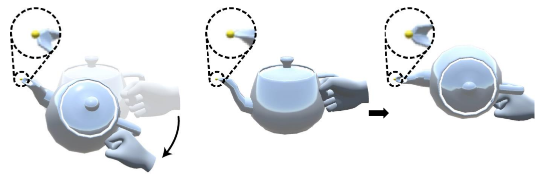


Fig. 7: Rotation about a pin: (a) 3DOF rotation. (b) Rolling the wrist.

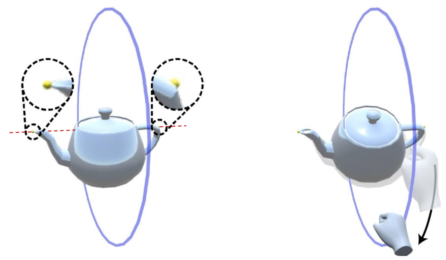


Fig. 8: 1DOF rotation with two locked pins: (a) The axis connecting the pins is only shown for illustration purposes. (b) Rotation with the `grab` gesture.

3.2.2 3DOF and 2DOF Rotations

A single locked pin fixes the part of the object bound to the pin in 3D space whereas the rest of the object can still be rotated. Acting as a pivot, the pin enables 3DOF and 2DOF rotations (shown in the second branch of Fig. 5). 3DOF rotation is made by touching the object with the `grab` gesture and then moving the hand (Fig. 7-(a)).

Consider the effect of rolling the wrist shown in Fig. 7-(b). Without it, object rotation is restricted to 2DOF. While performing 3DOF rotation, if the other hand also forms the `grab` gesture, the object is rotated in only 2DOF (ignoring wrist rotation) and the object's color changes to black.

Similar to the roles of 6DOF manipulation and 3DOF translation presented above, we expect that 3DOF rotation will be used to quickly approximate an object's orientation, followed by 2DOF rotation for precise alignment. To support this, 2DOF rotation is also scaled by $1/4$.

3.2.3 1DOF Rotation

In PinNPivot, locking a second pin maps to a more constrained rotation mode, i.e., the 1DOF rotation mode which appears in the third branch of Fig. 5. Two locked pins define a rotation axis, as shown in Fig. 8-(a). The direction of the rotation is indicated by a ring which appears upon locking two pins, with the ring being placed in the plane perpendicular to the rotation axis.

To perform 1DOF rotation, the user forms the `grab` gesture. This time, however, it is not necessary to touch the object. The user can rotate the object by moving their hand, as shown in Fig. 8-(b). 1DOF rotation is also scaled by $1/4$.

3.3 Interaction Example

The strength of PinNPivot is that it splits a challenging task into smaller and easier steps. Consider a *docking task*, where the *source* object needs to be manipulated to fit the *target* object that is fixed in 3D space. Fig. 9 shows a scenario where a docking task is completed with PinNPivot.

In Fig. 9-(a), the target object is displayed as semi-transparent and the source object is opaque. With PinNPivot, we recommend to place pins on corners or small protrusions of the object since these are usually easiest to accurately place and act well as pivots for the entire object. In the current example, the user's intention is to place a pin on the tip of the teapot's spout. To this end, 6DOF manipulation is used to bring the source object's spout close to the target's (Fig. 9-(b)) followed by scaled 3DOF translation to make the two spouts' tips coincide (Fig. 9-(c)). Then, a pin is created (Fig. 9-(d)) and locked (Fig. 9-(e)).

The user shifts their focus to another part of the teapot, in our example the handle. Using the locked pin as a pivot, the object is rotated in 3DOF (Fig. 9-(f)) and then in scaled 2DOF (Fig. 9-(g)) until the tips of the handles align.

Subsequently, the user creates a second pin (Fig. 9-(h)) and locks it (Fig. 9-(i)), which allows the user to rotate the object around the axis connecting the previous pin (locked in Fig. 9-(e)) and the new one. Finally, the object is rotated in scaled 1DOF until it matches the target (Fig. 9-(j)).

3.4 Design Rationale

PinNPivot's design evolved over time. In previous versions, users could perform 2DOF/1DOF translations and had access to unscaled 3DOF translation, scaled 3DOF rotation, and unscaled 2DOF/1DOF rotations. However, we noticed that these functions were not truly beneficial to achieving accuracy with our technique.

The core idea behind PinNPivot is to afford "a quick approximation followed by accurate fine tuning." In the first branch of Fig. 5, the quick approximation is provided through 6DOF manipulation, which is then followed by scaled 3DOF translation. In earlier versions of PinNPivot, support for unscaled 3DOF translation resulted in users unintentionally moving an already close object farther away from its target.

After one pin is locked (the second branch in Fig. 5), 3DOF rotation approximates the position of another point on the object and is followed by minuscule rotations. We found that the fine tuning of rotations is best achieved by reducing it to 2DOF and adding support for scaling.

Locking two pins implies that the user is almost satisfied with the object's orientation and therefore we provide scaled 1DOF rotation. We found that unscaled 1DOF rotation can result in unintentionally rotating the object farther away from its target in this phase.

In the earlier versions of PinNPivot, we tried various scaling factors. Similar to MAiOR [17], we found that $1/4$ was best for fine tuning.

4 Experiment

We recruited 20 participants (14 males and 6 females) aged 19 to 31 years (mean 25). Twelve participants had little to no experience (less than once a month) in VR and by extension little to no experience with the Oculus Rift headset and controllers. Ten participants had moderate experience

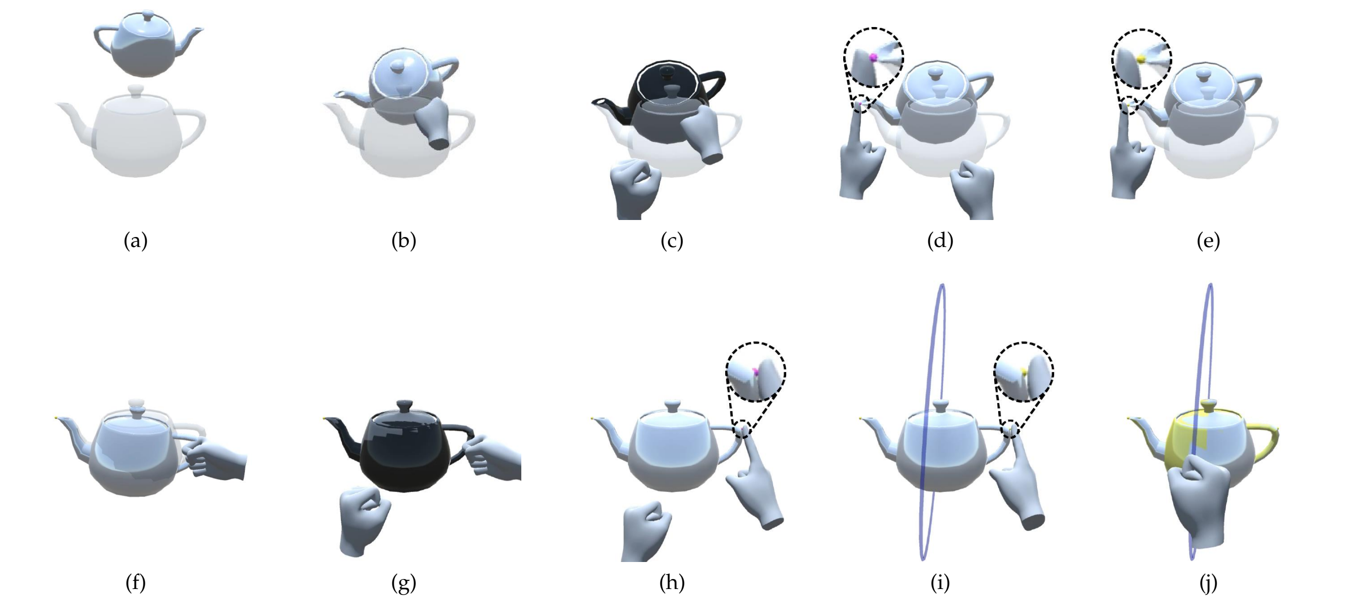


Fig. 9: A docking task completed with PinNPivot: (a) Source and target. (b) 6DOF manipulation quickly turns the object around. (c) Scaled 3DOF translation accurately places the spout's tip. (d) A pin is created. (e) It is locked. (f) The object is quickly rotated in 3DOF. (g) It is accurately rotated in scaled 2DOF. (h) A second pin is created. (i) It is locked and a ring appears. (j) The object is rotated in scaled 1DOF. When the target turns yellow, it indicates a good fit.

(several times a month) with 3D modeling software, such as Unity, whereas the rest had little to none. Four participants were left-handed and the remaining 16 were right-handed.

4.1 Baseline Techniques

We compared PinNPivot with three existing techniques presented in Section 2: (1) Direct 6DOF [1], the most intuitive, (2) SDOF [12], the most accurate, and (3) MAiOR [17], a recent technique that incorporates both scaling and DOF separation to improve accuracy. Readers are referred to Section 2 for their strengths and weaknesses. As mentioned in Section 3, PinNPivot is not affected by handedness, but the other techniques are. Thus, we adapted the techniques to each participant's handedness for the experiment.

4.1.1 6DOF

To manipulate an object the user simply grabs it with their dominant hand and moves it in 6DOF. The object then copies the translation and rotation of the hand.

4.1.2 SDOF

The SDOF technique manipulates an object through its primary axes. As shown in Fig. 10-(a), there are spheres at both ends of each axis. When the user grabs a sphere with their dominant hand and moves it along its axis, the object translates in 1DOF along the axis. To rotate the object, the user again grabs a sphere with their dominant hand, but this time moves it toward a sphere on another axis. The object then rotates in 1DOF about the third axis.

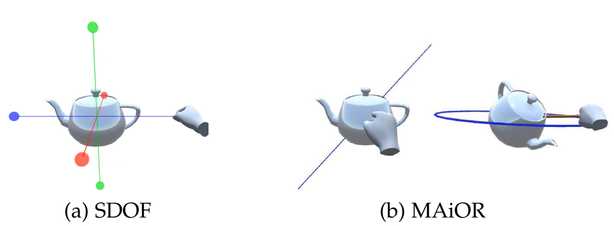


Fig. 10: SDOF and MAiOR used in the experiment.

4.1.3 MAiOR

In MAiOR, object rotation is initially disabled. Thus, if the user grabs the object with their dominant hand and moves it, the object translates in 3DOF. During 3DOF translation, an axis is displayed, connecting the object's starting position to its current position (Fig. 10-(b) left). The user can lock the axis to restrict the translation to 1DOF along that axis. Both 3DOF and 1DOF translations can be scaled by $^1/_4$.

For 3DOF rotation, the user forms the grab gesture with their dominant hand away from the object to first create a virtual bar. The user then connects the bar to the object, creating a lever with which they can rotate the object about its center in 3DOF. During 3DOF rotation, a ring is displayed (Fig. 10-(b) right). The center of the ring is that of the object, and its circumference passes through the lever. The user can lock the ring to restrict rotation to 1DOF.

4.2 Method and Procedure

Our experiment was made in a scene created with Unity. It had four objects: a painting, a plant, a couch, and a stuffed toy bunny. Fig. 11-(a) shows the first docking task, where the semi-transparent painting leans against the back wall while the source painting is to its left. Fig. 11-(b) shows the scene after the painting was placed and the second object to be manipulated (plant) is revealed. Fig. 11-(c) shows the third task (couch) and Fig. 11-(d) shows the fourth (bunny).

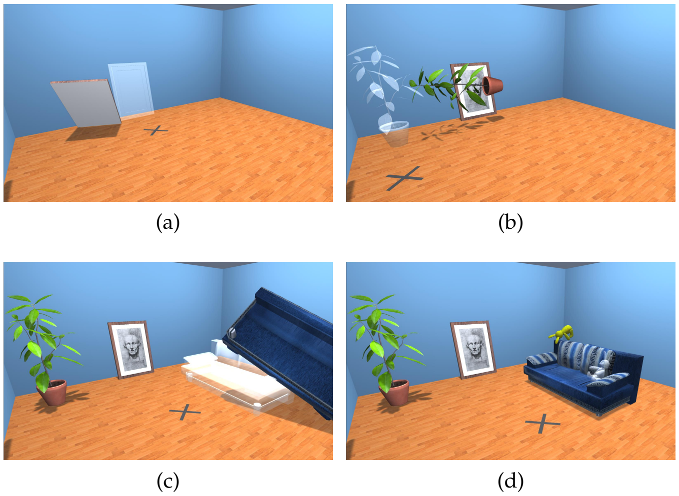


Fig. 11: The scene for the experiment: (a) Painting. (b) Plant. (c) Couch. (d) The target bunny is on the couch.

To mimic a real world application, all four docking tasks were designed to require full 6DOF transformations. When the source and target were within $1mm$ for position and $1°$ for rotation, the target turned yellow to indicate that the source was close enough and could be released. We measured placement accuracy after the source was released. We set the time limit for each docking task to 2.5 minutes as was done in the MAiOR experiment.

We wanted to investigate how object orientation and size affect the techniques' performances. Thus, the target plant and couch were orthogonally aligned with the walls and floor whereas the target painting and bunny were not. Also note that the four objects had substantially different sizes.

The experiment began with a general information questionnaire followed by a tutorial video showing the full capability of the first technique that the subject would be using. (The video for PinNPivot shows the steps in Fig. 9.) The subject was then set up in the IVE and given a five-minute practice session where they performed multiple teapot docking tasks. In the main experiment, a task ended when the time limit expired or if the user placed the object within the accuracy bounds. Then, the distance between the source's and target's centers was recorded as the positional error, and the angle between the two objects was recorded as the rotational error. After completing all four docking tasks, the subject filled a questionnaire about the technique they had used. Then, they repeated the process with the next technique, starting from the tutorial video. To reduce bias, we used a Latin Square to counterbalance the order of techniques in which each subject would complete the experiment.

5 Results and Discussion

As described above, we gathered objective (error and placement time) and subjective (questionnaire) data. The Shapiro-Wilk test indicated that the majority of data was non-normally distributed. Thus for our analysis, we used the Friedman non-parametric test and a Wilcoxon Signed-Rank post-hoc test with a Bonferoni correction.

We had three hypotheses for the outcome:

- PinNPivot leads to the lowest errors, due to its accuracy emphasis presented in Fig. 5.
- PinNPivot leads to the fastest placement time, due to its speed emphasis presented in Fig. 5.
- PinNPivot is considered the easiest-to-use, since it can break down a manipulation task into simple steps as described in Section 3.3.

In the subsequent subsections, the hypotheses will be discussed in order.

5.1 Positional and Rotational Errors

The mean positional/rotational errors for each of the four techniques for the four tasks are listed in Table 1. The top performer for each task is in bold font. PinNPivot scored the lowest errors for every task. Table 2 reveals that there are significant differences between errors, where p-values are presented using asterisks: $*$, $**$ and $***$ indicate < 0.05, < 0.01 and < 0.001 respectively. Fig. 12 shows the errors in terms of the median, interquartile ranges and 95% confidence. Also shown are pairwise statistical differences. PinNPivot had significantly lower errors than the other techniques for every task. This confirms our first hypothesis: PinNPivot leads to the lowest errors. As presented in Section 3.3 and Fig. 9, PinNPivot allows the user to focus on aligning a single part of an object at a time. We believe that this key characteristic gave PinNPivot the edge over the other techniques. After all, it is much easier to accurately place part of an object at a time than the entire object at once.

Table 1 also reveals a correlation between an object's size and positional error. The largest object (couch) had the largest positional errors for every technique, and the smallest object (bunny) had the smallest errors.

Due to the size of the couch, subjects were unable to see its entire shape while manipulating it. Based on observations during the study, we believe that all techniques

TABLE 1: Mean positional errors (in millimeters) and rotational errors (in degrees). In parentheses are the standard deviations. PNP stands for PinNPivot.

	positional error				rotational error			
object	**6DOF**	**SDOF**	**MAiOR**	**PNP**	**6DOF**	**SDOF**	**MAiOR**	**PNP**
painting	13.12(8.25)	9.29(8.58)	20.15(13.49)	**1.38(0.97)**	1.23(0.52)	0.95(0.93)	1.99(0.88)	**0.27(0.29)**
plant	11.09(6.50)	14.90(10.04)	16.59(12.01)	**1.73(1.42)**	1.91(1.78)	2.69(2.14)	2.88(1.74)	**0.27(0.18)**
couch	20.84(17.20)	33.10(32.17)	44.92(51.91)	**7.96(10.06)**	1.37(0.74)	1.87(1.65)	4.92(5.71)	**0.60(0.79)**
bunny	6.47(5.31)	7.90(4.79)	8.60(7.73)	**1.37(1.16)**	1.67(1.23)	2.37(1.23)	2.78(1.95)	**0.46(0.31)**

TABLE 2: Significant differences in positional/rotational errors discovered through Friedman's test.

object	positional error		rotational error	
	Friedman		Friedman	
	$X^2(3)$	p-value	$X^2(3)$	p-value
painting	26.49	$***$	28.20	$***$
plant	22.44	$***$	15.28	$**$
couch	19.62	$***$	22.22	$***$
bunny	26.37	$***$	32.94	$***$

TABLE 3: Percentages of subjects that completed the tasks (left). Significant differences in the task times discovered through Friedman's test (right - n.s. stands for no significance).

object	completion rates				task times	
	6DOF	SDOF	MAiOR	PNP	Friedman $X^2(3)$	p-value
painting	5%	0%	0%	**50%**	8.49	$*$
plant	0%	5%	0%	**50%**	8.49	$*$
couch	0%	0%	0%	**40%**	5.76	n.s.
bunny	20%	0%	25%	**75%**	18.02	$***$
average	6.25%	1.25%	6.25%	**53.75%**	27.60	$***$

suffered to varying degrees from this issue. Consider the 6DOF technique as an example. For a small object, the user can immediately see when and how their interactions affect the object. However, for a big object, the user can only realize the full scope of their manipulation by changing their view. Thus, the user might accurately manipulate a part of the object that they are looking at, but they cannot verify the rest of the object.

With PinNPivot, however, the object's size itself seemed to affect performance more than the visiblity issue. In a typical scenario with PinNPivot, a point of an object is first placed, pinned, and locked. Then the object is rotated until another point can be placed, pinned, and locked. If these manipulations are successfully performed, both orientation and position are simultaneously matched with 1DOF rotation. If even one of the points were incorrectly placed, however, the user would have to unlock, reposition, re-lock and attempt 1DOF rotation again. The bigger the object is, the longer it takes to walk around during adjustments. Given the time limit in the experiment, this resulted in the largest positional and rotational errors for the couch.

Unlike positional error, rotational error is much less affected by object size. For example, no technique had the smallest error for the smallest object (bunny) and only two (PinNPivot and MAiOR) had the largest errors for the largest object (couch). Somewhat unsurprisingly, since corners are easier to accurately place, PinNPivot's rotational errors are smaller on objects that have sharp corners (painting and plant) and bigger on "rounder" objects (bunny).

Recall that the target poses of the plant and couch were orthogonally aligned, whereas they were not for the painting and bunny. Table 1 shows that such orientation differences did not affect the performances of the techniques.

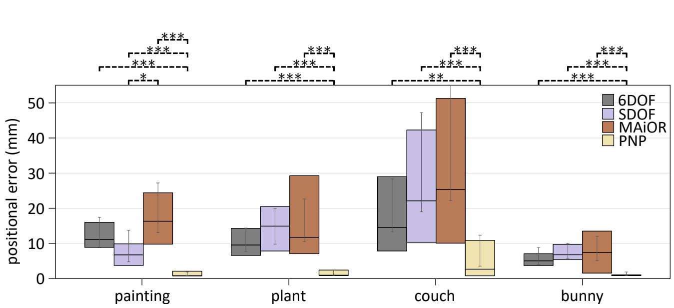
5.2 Task Times

A task ended either when the subject placed the object within the accuracy bounds or when time ran out. Table 3 shows the percentages of subjects that were able to place the objects within the time limit. On average, half of the subjects were able to complete the tasks with PinNPivot, but much fewer could finish with the other techniques.

Fig. 13 shows a dot plot and statistical differences for (a) individual task times and (b) the subjects' average task times. A dot at 150 seconds (2.5 minutes) indicates that the subject timed out. Dots that are more opaque indicate a pile-up of identical results. Fig. 13 and Table 3 show that for both the individual task times (except with the couch) and average task times, PinNPivot was significantly faster.

As mentioned earlier, a useful object manipulation technique must be both accurate and fast. When designing our experiment, we had to carefully balance the accuracy and time limits to make the tasks challenging with PinNPivot without prompting participants to simply give up with the other techniques. Through pilot studies, we found that longer time limits resulted in subjects stopping, either out of fatigue or frustration. If a subject "gives up" midway through a task, measuring and comparing task times becomes complicated. Thus, we encouraged subjects to continue trying until the 2.5 minutes ran out. In our experiment, no subjects gave up.

The downside of the 2.5-minute time limit is that the upper distribution of task times were cut off. However, with PinNPivot, 53.75% of all tasks were finished within the accuracy limits, while only 6.25% were completed with 6DOF and MAiOR. Although we cannot identify a second



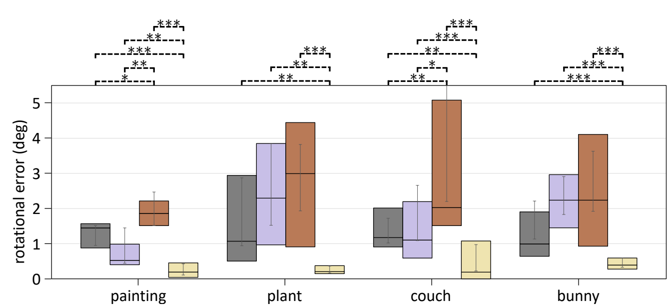


Fig. 12: Positional errors (left) and rotational errors (right). Presented are the median, interquartile ranges, and 95% confidence (whiskers). Dotted brackets indicate pairwise statistical differences.

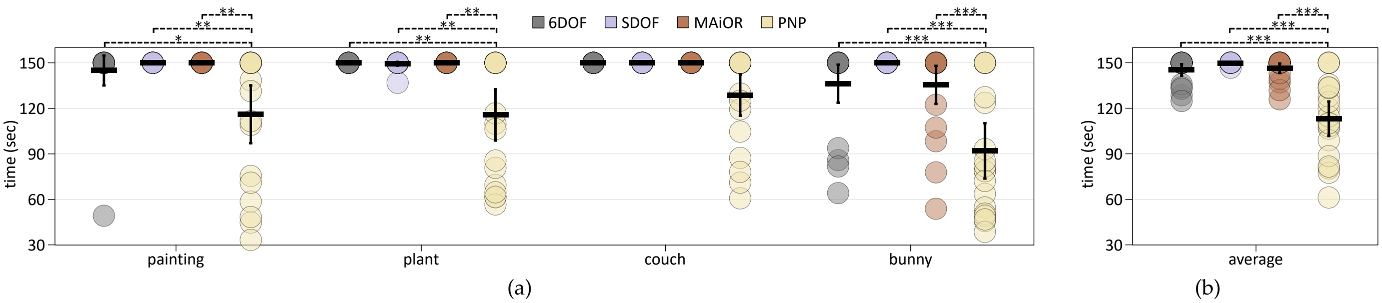


Fig. 13: A dot plot of the subjects' task times with the mean (horizontal bar) and 95% confidence (whiskers): (a) Four techniques' task times per object. (b) Average task times.

TABLE 4: The questionnaire and significant differences in responses discovered through Friedman's test.

#	question	Friedman $X^2(3)$	Friedman p-value
1	The technique is easy to understand.	24.11	$***$
2	The technique is easy to use.	9.56	$*$
3	Translation is easy to do.	20.30	$***$
4	Rotation is easy to do.	21.90	$***$
5	The objects reacts as I expect.	29.66	$***$
6	I am able to accurately translate objects.	13.61	$**$
7	I am able to accurately rotate objects.	17.84	$***$
8	The technique is tiring.	8.99	$*$
9	The technique is fun.	21.52	$***$
10	More practice will improve my ability to use the technique.	10.19	$*$

fastest technique, we can still confirm our second hypothesis: PinNPivot leads to the fastest placement time.

For PinNPivot we observed that the majority of time was spent on placing the first pin, i.e., the pivot for 3DOF rotation. Since subsequent manipulations depend on that point, subjects naturally attempted to perfect it before moving on.

5.3 Questionnaires and Responses

The questionnaire and Friedman's test results are shown in Table 4. Responses (on a 7-point Likert scale) are shown in Fig. 14. The responses to question 1 show that 6DOF and SDOF were considered more intuitive than PinNPivot. This was expected since 6DOF mimics real world interactions and 3D modeling software is generally similar to how SDOF works (recall that half of the subjects had moderate experience with 3D modeling software). Nonetheless, PinNPivot was also found to be intuitive, which implies that our metaphor of "pinning" is an appropriate choice.

Questions 2, 3 and 4 investigated easiness-of-use, and our last hypothesis ("PinNPivot is considered the easiest-to-use") was confirmed. The subjects also found 6DOF easy but were undecided on SDOF. MAiOR was perceived as not easy-to-use. Looking at questions 3 and 4 separately, we see that MAiOR suffered from problems with rotation, which is what the original authors found.

Questions 5, 6 and 7 addressed controllability and accuracy. Here, PinNPivot scored the highest. The subjects felt that SDOF provided some control but did not consider it accurate. Similar observations were made for 6DOF.

Questions 8 and 9 dealt with fatigue and fun. PinNPivot was found to be non-tiring and fun, whereas the other techniques yielded the opposite results. In general, both 6DOF and MAiOR required trial and error. In 6DOF, for example, the user rapidly opened and closed their hand multiple times to reduce the effect of a single manipulation. With SDOF, one subject commented "Placing the source object near to the target is not hard, but accurate modification is." This is because accuracy requires the user to perform multiple minuscule 1DOF adjustments and it is not always obvious whether rotation or translation is needed.

Question 10 asked whether additional practice might improve the users' ability with each technique. Subjects agreed that they would improve with PinNPivot but would not improve with 6DOF. They were undecided about whether they could improve with SDOF and MAiOR.

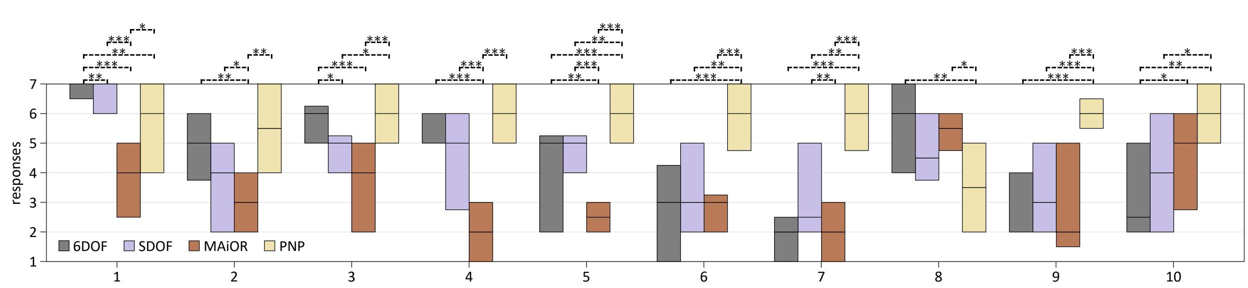


Fig. 14: Median and interquartile ranges for the responses to the questionnaire (on a 7-point Likert scale).

6 Conclusion and Future Work

In this paper we proposed a new object manipulation technique called PinNPivot. The technique uses pins to restrict the rotation of the object. To evaluate our technique we compared it against three previous techniques: 6DOF, SDOF and MAiOR. According to the results, PinNPivot was statistically more accurate than the other techniques in both translation and rotation. PinNPivot was also significantly faster. Participants also found PinNPivot to be easier, less tiring and more fun than the others.

As a future work, we plan to continue testing PinNPivot in different environments, e.g., with even larger variations of object sizes, shapes and orientations. We would also like to evaluate the effect of longer-term practice on PinNPivot. Finally, we plan to add uniform object scaling and compare this enhanced technique against existing 7DOF ones.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korea government (MSIT) (NRF-2017M3C4A7066316 and No. NRF2016-R1A2B3014319).

References

[1] R. Wang, S. Paris, and J. Popović, “6d hands: markerless hand-tracking for computer aided design,” in *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, 2011, pp. 549–558.

[2] D. A. Bowman and L. F. Hodges, “An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments,” in *Proceedings of the 1997 symposium on Interactive 3D graphics*. ACM, 1997, pp. 35–ff.

[3] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa, “The go-go interaction technique: non-linear mapping for direct manipulation in vr,” in *Proceedings of the 9th annual ACM symposium on User interface software and technology*. ACM, 1996, pp. 79–80.

[4] J. S. Pierce, B. C. Stearns, and R. Pausch, “Voodoo dolls: seamless interaction at multiple scales in virtual environments,” in *Proceedings of the 1999 symposium on Interactive 3D graphics*. ACM, 1999, pp. 141–145.

[5] B. R. De Araújo, G. Casiez, J. A. Jorge, and M. Hachet, “Mockup builder: 3d modeling on and above the surface,” *Computers & Graphics*, vol. 37, no. 3, pp. 165–178, 2013.

[6] T. Kim and J. Park, “3d object manipulation using virtual handles with a grabbing metaphor,” *IEEE Computer Graphics and Applications*, vol. 34, no. 3, pp. 30–38, 2014.

[7] S. Frees, G. D. Kessler, and E. Kay, “Prism interaction for enhancing control in immersive virtual environments,” *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 14, no. 1, p. 2, 2007.

[8] D. Mendes, F. M. Caputo, A. Giachetti, A. Ferreira, and J. Jorge, “A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments,” in *Computer Graphics Forum*, vol. 38, no. 1. Wiley Online Library, 2019, pp. 21–45.

[9] C. Wilkes and D. A. Bowman, “Advantages of velocity-based scaling for distant 3d manipulation,” in *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*. ACM, 2008, pp. 23–29.

[10] N. Osawa, “Two-handed and one-handed techniques for precise and efficient manipulation in immersive virtual environments,” in *International Symposium on Visual Computing*. Springer, 2008, pp. 987–997.

[11] B. Bossavit, A. Marzo, O. Ardaiz, L. D. De Cerio, and A. Pina, “Design choices and their implications for 3d mid-air manipulation techniques,” *Presence: Teleoperators and Virtual Environments*, vol. 23, no. 4, pp. 377–392, 2014.

[12] D. Mendes, F. Relvas, A. Ferreira, and J. Jorge, “The benefits of dof separation in mid-air 3d object manipulation,” in *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. ACM, 2016, pp. 261–268.

[13] P. Song, W. B. Goh, W. Hutama, C.-W. Fu, and X. Liu, “A handle bar metaphor for virtual object manipulation with mid-air interaction,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 1297–1306.

[14] I. Cho and Z. Wartell, “Evaluation of a bimanual simultaneous 7dof interaction technique in virtual environments,” in *3D User Interfaces (3DUI), 2015 IEEE Symposium on*. IEEE, 2015, pp. 133–136.

[15] F. M. Caputo, M. Emporio, and A. Giachetti, “The smart pin: an effective tool for object manipulation in immersive virtual reality environments,” *Computers & Graphics*, 2018.

[16] T. T. H. Nguyen, T. Duval, and C. Pontonnier, “A new direct manipulation technique for immersive 3d virtual environments,” in *ICAT-EGVE 2014: the 24th International Conference on Artificial Reality and Telexistence and the 19th Eurographics Symposium on Virtual Environments*, 2014, p. 8.

[17] D. Mendes, M. Sousa, R. Lorena, A. Ferreira, and J. Jorge, “Using custom transformation axes for mid-air manipulation of 3d virtual objects,” in *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. ACM, 2017, p. 27.

P. Christopher Gloumeau Paul Christopher Gloumeau obtained a B.S. degree in computer science at The University of the West Indies, Cave Hill Campus after which he was awarded an all inclusive scholarship to pursue his master’s degree. While completing his M.Eng. degree in computer science, he joined the Interactive 3D Media Lab where he researched virtual reality and augmented reality for 2 years.

Wolfgang Stuerzlinger Building on his deep expertise in virtual reality and human-computer interaction, Dr. Stuerzlinger is a leading researcher in spatial and three-dimensional user interfaces. Since 2014, he is a full professor at the School of Interactive Arts + Technology at Simon Fraser University in Vancouver, Canada. Current research projects include better 3D interaction techniques for VR and AR applications, new human-in-the-loop systems for big data analysis (visual analytics and immersive analytics), the characterization of the effects of technology limitations on human performance, investigations of human behaviors with occasionally failing technologies, user interfaces for versions, scenarios and alternatives, and new virtual reality hardware and software.

JungHyun Han Dr. JungHyun Han is a professor in the Computer Science Department and the director of the Interactive 3D Media Lab at Korea University. He is also directing the Next-generation Game Research Center founded by the Korean government. Dr. Han received a B.S. degree from Seoul National University and a Ph.D. degree from the University of Southern California. Prior to joining Korea University, he worked at the US Department of Commerce National Institute of Standards and Technology.