

EZ-Manipulator: Designing a mobile, fast, and ambiguity-free 3D manipulation interface using smartphones

Po-Huan Tseng¹, Shih-Hsuan Hung² (✉), Pei-Ying Chiang³, Chih-Yuan Yao², and Hung-Kuo Chu¹

© The Author(s) 2018. This article is published with open access at Springerlink.com

Abstract Interacting with digital contents in 3D is an essential task in various applications such as modeling packages, gaming, virtual reality, etc. Traditional interfaces using keyboard and mouse or trackball usually require a non-trivial amount of working space as well as a learning process. We present the design of EZ-Manipulator, a new 3D manipulation interface using smartphones that supports mobile, fast, and ambiguity-free interaction with 3D objects. Our system leverages the built-in multi-touch input and gyroscope sensor of smartphones to achieve 9 degrees-of-freedom axis-constrained manipulation and free-form rotation. Using EZ-Manipulator to manipulate objects in 3D is easy. The user merely has to perform intuitive single- or two-finger gestures and rotate the hand-held device to perform manipulations at fine-grained and coarse levels respectively. We further investigate the ambiguity in manipulation introduced by indirect manipulations using a multi-touch interface, and propose a dynamic virtual camera adjustment to effectively resolve the ambiguity. A preliminary study shows that our system has significant lower task completion time compared to conventional use of a keyboard–mouse interface, and provides a positive user experience to both novices and experts.

Keywords 3D manipulation interface; interaction technique; multi-touch; gyroscope sensor; mobile device; input device

1 Introduction

3D object manipulation is a key task in many applications, including virtual reality, gaming, and commercial modeling packages such as 3DS Max, Maya, etc. Manipulation is typically achieved through a traditional keyboard mouse interface using a virtual trackball [1, 2] or a 3D transformation widget [3, 4]. Researchers have done extensive work on designing novel input devices with more degrees of freedom (DOF) [5–8] and mid-air interaction using hand gestures [9–11]. However, those systems usually demand specialised hardware (e.g., depth sensors [10, 11]) and require a non-trivial environment. For instance, most mouse-like devices need to be placed or moved on a desk, while a depth sensor is capable of tracking hand gestures only within a fixed range of distances. These requirements largely limit the usability and mobility of the above systems.

Motivated by the rapid development of mobile devices with multi-touch interfaces, a huge body of work has been proposed to adapt tactile paradigms to 3D manipulation [12–22]. Amongst existing systems, indirect manipulation techniques [17, 18] that disassociate the transformations from visual widgets have been shown to be easier to use than direct ones [20, 21]. However, they assume multi-touch devices with mid- to large-sized displays (e.g., a tablet or touch-table) and hence the proposed two- or three-finger gestures used are unsuitable for touch screens of small size as found on a smartphone. Naive adaptation of these methods will lead to “fat finger” and hand occlusion problems. Moreover, indirect manipulation using multi-touch gestures frequently causes *ambiguous manipulation*, where the 2D projection of a 3D transformation widget results in

1 “National Tsing Hua University”, Hsinchu 30013, Taiwan, China. E-mail: P.-H. Tseng, x053125@gmail.com; H.-K. Chu, hkchu@cs.nthu.edu.tw.

2 “National Taiwan University of Science and Technology”, Taipei 10607, Taiwan, China. E-mail: S.-H. Hung, m10215089@csie.ntust.edu.tw (✉); C.-Y. Yao, cyuan.yao@csie.ntust.edu.tw.

3 “National Taipei University of Technology”, Taipei 10607, Taiwan, China. E-mail: peiyingc@csie.ntut.edu.tw.

Manuscript received: 2017-12-25; accepted: 2017-12-30

overlapping coordinate axes on the 2D screen. Thus, while multi-touch interfaces are promising for 3D manipulation, we still lack a feasible approach for smartphones.

In this work, we present EZ-Manipulator, a new 3D manipulation interface that leverages the built-in multi-touch input and gyroscope sensor in a smartphone to support mobile, fast, and ambiguity-free 9 DOF axis-constrained manipulation, with 3 DOF each for translation, rotation and scaling, and free-form rotation. Specifically, we base our work on the work of Au et al. [18] to achieve 9 DOF axis-constrained manipulation through a small set of multi-touch gestures using intuitive single- and two-finger operations (see Fig. 1(a)). To tackle ambiguous manipulation, we develop a *dynamic camera* system that monitors the 2D projection of the 3D transformation widget and automatically adjusts the viewpoint of the virtual camera to resolve ambiguity. To freely rotate 3D objects, our system virtually associates the coordinate system of the object with that of the physical smartphone, and directly controls the object's orientation using the reference orientation from the gyroscope sensor (see Fig. 1(b)). The handiness and wireless connection capability of smartphones further enable a scenario of mobile and remote interaction with 3D applications using EZ-Manipulator (see Fig. 1(c)).

A preliminary study shows that our system not only outperforms conventional keyboard–mouse interfaces in terms of task completion time, but also that its efficacy and intuitiveness are highly valued by both novices and experts. We have also demonstrated the usability of EZ-Manipulator in several application scenarios, including remote interaction with a large display environment, rapid prototyping of interior

designs, multi-user collaboration, and gaming (see the video in the Electronic Supplementary Material (ESM)).

2 Related work

2.1 Multi-touch interaction

The rapid development of multi-touch display devices has motivated much work on the design of 2D multi-touch gestures for 3D manipulation tasks [12, 13, 15–18, 21–24]. For instance, Hancock et al. [12, 13] explored how to map multi-touch inputs to 6 DOF unconstrained 3D manipulation, i.e., translation and free-form rotation. Cohé et al. [21] presented tBox, a novel box-like widget designed for touch screens to facilitate 3D interaction using a 2D widget. Au et al. [18] further achieved a widget- and button-free constrained 3D manipulation interface using a small but effective set of multi-touch gestures. Beyond basic 3D manipulation, some work utilizes multi-touch input in specific contexts, including 3D modeling [23, 24], organic environment construction [22], automation engineering [25], and stereo workspace construction [19]. However, all of these systems require a large display to support interaction with the underlying applications, and hence are largely limited in mobility. A naive adaptation to the smartphone paradigm (with a small display) leads to “fat finger” and hand occlusion problems. In contrast, our work carefully adapts the multi-touch gestures in Ref. [18] to smartphones, and leverages the built-in sensors to enable both constrained and unconstrained 3D manipulations with 9 DOF.

2.2 Mid-air interaction

Another line of work proposes to interact with 3D

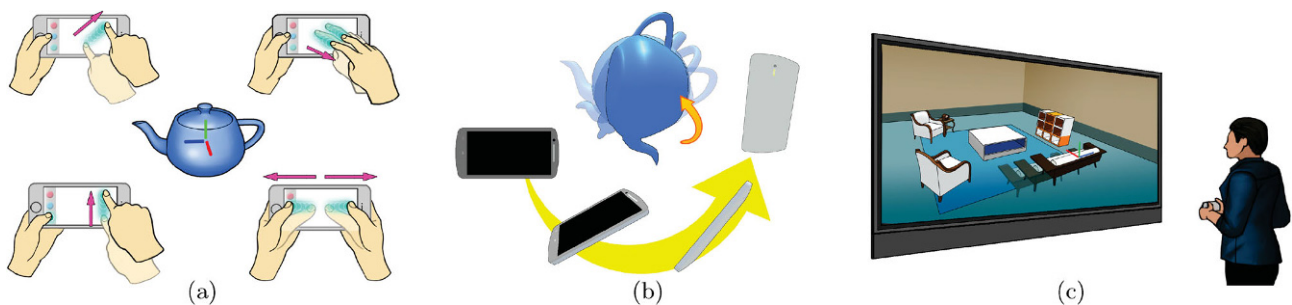


Fig. 1 EZ-Manipulator is a new 3D manipulation interface for smartphones that supports mobile, fast, and ambiguity-free interaction with 3D objects. The system utilizes (a) a small set of intuitive multi-touch gestures to achieve 9 DOF axis-constrained manipulation, and (b) performs free-form rotation using the built-in gyroscope sensor. (c) A user remotely manipulating 3D objects on a large-size display using EZ-Manipulator.

content using natural freehand or body gestures [10, 11, 26, 27]. Wang et al. [10] presented a markerless hand tracking system with a set of tailor-made gestures for performing fundamental tasks in CAD applications. Song et al. [11] introduced a handle bar metaphor to provide effective and accurate control over pose and scale manipulations of 3D objects. Nancel et al. [27] designed a set of mid-air gestures to support pan-and-zoom interaction with digital content shown on a wall-sized display. Hilliges et al. [26] leveraged the space above a tabletop display to support intuitive and direct 3D manipulations using a see-through display. However, these tracking-based systems are functional only in non-trivial settings where users have to perform the gestures within a working area sensed by an RGB-D camera [28] or when wearing a colored glove [9], as means of hand tracking and gesture recognition. In addition, performing mid-air gestures can cause fatigue and is hence unsuitable for long term usage. Our system overcomes the above issues by using an ubiquitous smartphone for the interface, which takes negligible effort to use.

2.3 Interaction using mobile devices

Some work has shown the advantages of using mobile devices in remote interaction and collaboration [29–33]. Boring et al. [32] introduced the Touch Projector, an interface implemented on a mobile device that allows users to manipulate contents shown on a distant display. The system has been further extended to support interaction of multiple users and has been applied to media facades through live video [31]. Chapuis et al. [29] presented Smarties, an input interface using a mobile device that supports complex interaction and collaboration among users for wall-sized display applications. Chang et al. [33] implemented object tracking and annotation on a mobile device to support remote collaboration in a dynamic video-mediated workspace. Similarly, our system utilizes the mobility and handiness of a smartphone to support intuitive and effective 3D manipulation.

3 System overview

Since the EZ-Manipulator is designed to enable single or multiple users to manipulate 3D objects on a remote display, we employ a client–server protocol

to communicate between the smartphones (clients) and virtual environment (server). EZ-Manipulator has three modes of use in 3D modeling. In the key *manipulation* mode, users may perform both fine-grained and coarse 3D manipulations of virtual objects through intuitive multi-touch gestures, and by rotating the smartphone by hand. Without loss of generality, we assume the dominant hands of users to be their right hands. The other two modes are *camera* and *selection* modes, in which users may respectively control the camera view and select single or multiple 3D objects. Users can seamlessly switch modes, thanks to the small touch screen of the smartphone, using different gesture patterns.

4 Interaction design

4.1 Multi-touch gestures: axis-constrained manipulation

Indirect control of a 3D transformation widget using multi-touch gestures has been proved to be effective for constrained manipulation of 3D objects [18]. As our interface shares the same idea of indirect manipulation using multi-touch gestures, we thus design a visual transformation widget for each 3D object. Given an object selected for editing, we define candidate axes for the object using its local coordinate frame and render these axes in distinguishable colors to represent a visual transformation widget drawn at the centroid of the object. Next we base on the following observations to re-design the multi-touch gestures for smartphones.

4.1.1 One finger is better than two

A pilot study by Kashiwakuma et al. [34] indicated that single-finger pan offers more precise control than using two fingers, and is preferred by most users. Furthermore, a single finger, which touches a smaller area, can pan more freely on a small touch screen than two fingers. Thus we base gestures on single-finger interactions.

4.1.2 Locations of touched points matter

It is easy to show that when we hold the smartphone with one or two hands, the right (or left) thumb can easily touch regions near the right (or left) border of the screen. This enables us to increase the dimensionality of multi-touch gestures based on the locations of the touched points.

Figure 2 illustrates the design of multi-touch

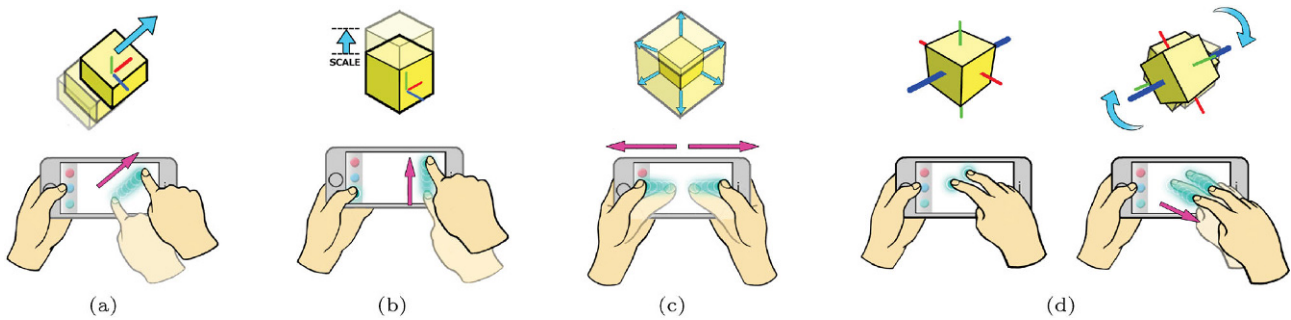


Fig. 2 Multi-touch gestures for 9 DOF axis-constrained manipulation, including (a) translation, (b) non-uniform scaling, (c) uniform scaling, and (d) rotation.

gestures for 9 DOF axis-constrained manipulation. We divide the touch screen into two regions, one for triggering finger touch events and the other for detecting single- and two-finger pan and pinch gestures. The former region, the *TouchZone*, is a vertical rectangular area aligned with the left border of the touch screen. We roughly classify the location of touched points within the *TouchZone* into three areas, top, middle, and bottom, which are visualized using circular shapes. Therefore we obtain 12 combinations in total using 4 finger touching states from the non-dominant hand and 3 finger movement patterns from the dominant hand.

To control the axis-constrained transformations using a single-finger pan gesture, our system automatically compares its movement direction with 2D lines, which represent the projections of candidate axes on the view plane, and selects the one with smallest angular difference as the candidate axis. The magnitude of the transformation is calculated from the displacement. Specifically, axis-constrained translation is achieved by the gesture consisting of a single-finger pan along the desired axis (see Fig. 2(a)). Axis-constrained scaling is performed by two-handed gesture that combines a single-finger touch on the bottom of the *TouchZone* and a single-finger pan along the desired axis (see Fig. 2(b)). Uniform-scaling is triggered using a direction-independent two-finger pinch gesture (see Fig. 2(c)). Finally, we employ the same design as Ref. [18] to perform axis-constrained rotation using the gesture of a two-finger pan perpendicular to the desired axis (see Fig. 2(d)).

4.2 Gyroscope: free-form rotation

In augmented reality applications, the advantage of using the mobile device's sensors for interaction with

3D content is that the 3D scene can be viewed and controlled intuitively by rotating and moving the physical device. This inspires us to base the free-form rotation control on physical smartphones in such a way that users can freely control object orientation simply by rotating the phone by hand. This can be done by associating the object orientation with the reference orientation data from the built-in gyroscope sensor (see Fig. 3).

4.3 Camera control and object selection

Camera control is a basic but essential operation for users to see 3D objects from different viewpoints. To offer a basic control mechanism, we constrain the camera view to always be centered on the centroid of the manipulated object, while users can rotate the camera using a two-handed gesture that combines a single-finger touch on top of the *TouchZone* and a single-finger pan (see Fig. 4(a)).

Another frequently used operation is to select one or more objects in the virtual environment. We use a two-handed gesture similar to the one used for camera control, except that the single-finger touch is now on the middle of the *TouchZone*, while the single-finger pan gesture moves a virtual cursor in 2D (see Fig. 4(b)). The system automatically highlights each object pointed to by the virtual cursor; users

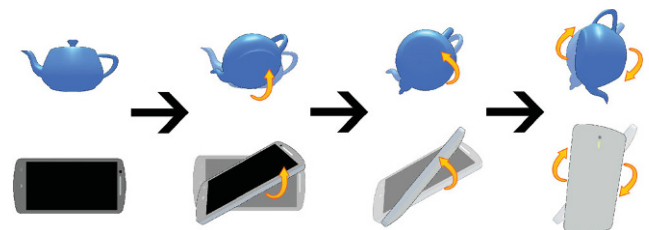


Fig. 3 Free-form rotation using a reference rotation inferred from the gyroscope sensor.

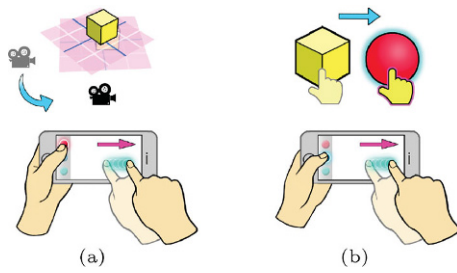


Fig. 4 Multi-touch gestures for (a) camera control and (b) object selection.

can confirm the selection by releasing the touch finger on the TouchZone.

5 Ambiguous manipulation

5.1 Ambiguity

Determination of a unique candidate axis is essential for valid axis-constrained manipulation. In our system, this task is done by comparing the orientation of 2D lines, the projections of candidate axes on the view plane, with either the direction of a single-finger pan gesture (for translation or non-uniform scaling) or the orientation defined by two touch points (for rotation). When 2D lines are nearly collinear in the view plane, the system can not effectively compute a unique candidate axis, and we call such a scenario an ambiguous manipulation (see Fig. 5(a)). Although one can resolve the ambiguity by altering the camera view, this extra effort will typically slow down the manipulation task. We thus provide a simple yet effective system to automatically resolve the ambiguity problem. The key insight lies in formulating an optimization problem that finds a new camera view which is ambiguity-free and does not deviate too much from the original view.

5.2 Dynamic camera adjustment

Given a camera view C_i , we denote the projection of candidate axes on the view plane of C_i as a triple

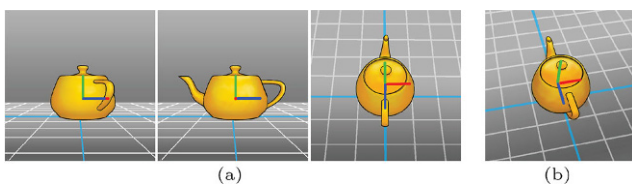


Fig. 5 (a) Examples of ambiguous configurations. (b) The ambiguity on the right in (a) is resolved using dynamic camera adjustment.

$\{\mathbf{X}_i, \mathbf{Y}_i, \mathbf{Z}_i\}$. To detect the occurrence of ambiguity, we use the following delta function:

$$\delta(C_i) = \begin{cases} 1, & \text{if } \min\{f_{\mathbf{X}\mathbf{Y}}^i, f_{\mathbf{X}\mathbf{Z}}^i, f_{\mathbf{Y}\mathbf{Z}}^i\} \leq 10^\circ \\ 0, & \text{otherwise} \end{cases}$$

where $f_{\mathbf{X}\mathbf{Y}}$ is the angle between the 2D vectors \mathbf{X} and \mathbf{Y} . We measure how far the current camera view is from an ambiguous configuration using the equation:

$$E_a(C_i) = \max\{f_{\mathbf{X}\mathbf{Y}}^i, f_{\mathbf{X}\mathbf{Z}}^i, f_{\mathbf{Y}\mathbf{Z}}^i\} - \min\{f_{\mathbf{X}\mathbf{Y}}^i, f_{\mathbf{X}\mathbf{Z}}^i, f_{\mathbf{Y}\mathbf{Z}}^i\}$$

Since the camera view is constrained to be centered on the object, we model the distance between the two camera views as

$$E_d(C_i, C_j) = \|\mathbf{p}_i - \mathbf{p}_j\|$$

where \mathbf{p}_i is the 3D position of C_i . Overall, if $\delta(C_i) \neq 0$, our goal is to find a new C'_i that minimizes the objective function:

$$E(C_i, C'_i) = E_a(C'_i) + E_d(C_i, C'_i) + 100 \cdot \delta(C'_i) \quad (1)$$

With proper normalization of the first two terms in Eq. (1), we find a local minimum by randomly sampling around the current camera view (15 samples) and taking the one with best value.

6 Evaluation

Three experimental studies were conducted to evaluate the performance of EZ-Manipulator. In the first experiment, we compared our interface with a conventional keyboard–mouse interface by measuring the completion time for a series of 3D manipulation tasks executed by the participants.

In the second experiment, we evaluated the effectiveness of our proposed dynamic camera adjustment for ambiguous manipulations with a set of tailor-made 3D manipulation tasks.

Finally we collected user experiences from both novices and experts in the third experiment to evaluate the usability of our system.

We recruited thirteen novice users, including 11 males and 2 females with mean age of 23, who participated in all experiments. Further five experts including 1 male and 4 females with mean age of 25 participated in the third experiment. No novice user had prior experience of 3D manipulation, while all experts had at least three years of experience in 3D modeling packages such as 3DS Max or Maya. All the participants were well-trained smartphone users.

6.1 3D object manipulation

In Experiment I, each participant was asked to accomplish 12 tasks of adjusting the pose of a 3D object to a specified configuration using both a keyboard–mouse interface and EZ-Manipulator. The total task completion time for each participant was recorded. Note that each re-posing task was specifically designed so that participants would have to perform translation, rotation, and scaling operations to finish the task. Each participant was given 5 training tasks using both interfaces before starting the assessed tasks. Results are shown in Fig. 6(a). They indicate that our interface is much more efficient than a conventional keyboard–mouse interface with significant reduction in task completion time.

6.2 Ambiguous manipulation

Experiment II was conducted to evaluate the effectiveness of our dynamic camera adjustment system in helping users to resolve ambiguous manipulations. We prepared a set of 10 object re-posing tasks similar to those in the previous experiment. Each task involved only translation and was designed to guarantee more than one occurrence of ambiguity while finishing the task. Each participant was asked to accomplish 10 tasks using our interface with and without the assistance of dynamic camera adjustment, and we recorded the total completion time for each participant. We can see from Fig. 6(b) that the dynamic camera did indeed help reduce the total task completion time for 3D object manipulations. Our observations showed that participants typically spent much time adjusting camera views when dynamic camera adjustment was unavailable.

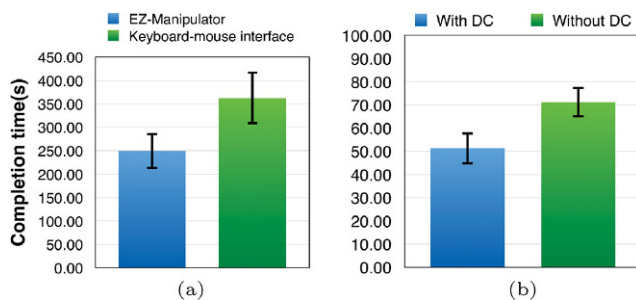


Fig. 6 Results for (a) Experiment I and (b) Experiment II.

6.3 Usability

Experiment III comprised a set of questionnaires given to each participant after the two previous user studies, to evaluate the usability of EZ-Manipulator for both experts and novices. The mean response of the 17 participants and associated standard deviation bar can be found in Fig. 7. Responses from novices were in general higher than those from experts, especially for the opinion “a dynamic camera is helpful”. The responses from the experts indicate they can readily resolve the ambiguity using a combination of keyboard and mouse input, and for them, the dynamic camera system is unnecessary. A further common opinion voiced by participants is that the axis-constrained rotation could not achieve the demanded precision using a two-finger gesture. Overall, however, both experts and novices agree that EZ-Manipulator is a fast, easy, and intuitive 3D manipulation interface that is particularly suitable for fast prototyping.

7 Applications

We further used EZ-Manipulator in three different applications to illustrate its potential (see Fig. 8). The first application demonstrates how the user can quickly re-arrange the furniture in an interior design using our system. The second application shows the advantage of using our system to support remote interaction and multi-user collaboration (two users collaborate to re-arrange furniture) on a wall-sized display. The third application implements a multi-player Jenga game in which the user can easily pick a piece and remove it using the axis-constrained manipulations offered by EZ-Manipulator. We refer the reader to the supplementary video in the ESM for demonstrations.

8 Limitations

EZ-Manipulator has several limitations. (i) Because manipulation is performed via the multi-touch interface on a small-screen device, accurate manipulation requires tiny finger movements, which are difficult and time-consuming. (ii) During multi-user collaboration, camera control can cause problems as they share a single camera on the workspace. If one user controls the camera, it will immediately impact the other users. To a certain extent, our dynamic

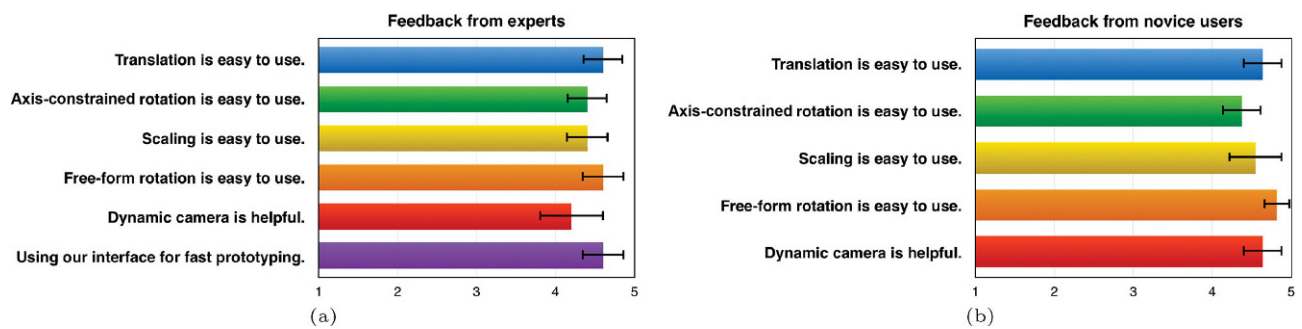


Fig. 7 User ratings on a Likert scale from 1 (strongly disagree) to 5 (strongly agree) for each user study.

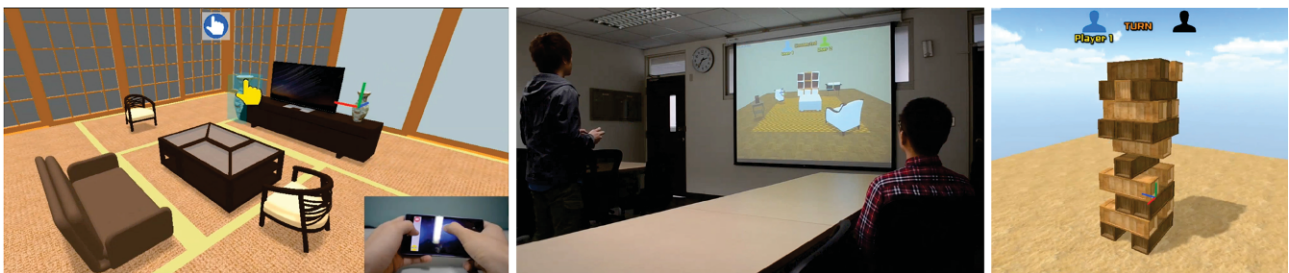


Fig. 8 Applications using EZ-Manipulator. Left: rapid prototyping of interior designs. Centre: multiple user collaboration and interaction using a wall-sized display. Right: multi-player gaming.

camera also has this problem. Therefore, we do not permit dynamic camera operations when multiple users are manipulating objects in a scene.

9 Conclusions and future work

In this paper, we have proposed EZ-Manipulator, a new 3D manipulation interface using smartphones that provides 9 DOF axis-constrained manipulation and free-form rotation. To overcome ambiguous manipulation problems in our interface, we have implemented a dynamic camera, which can effectively resolve the ambiguity. According to user feedback, our system provides a positive user experience for both novice users and experts. Results of our user study indicate that our interface is faster than a keyboard–mouse interface for novice users. In future, we hope to enrich and improve our interface according to the feedback from experts, while keeping the advantages of the current interface. Moreover, we hope to investigate further applications based on EZ-Manipulator.

Electronic Supplementary Material Supplementary material is available in the online version of this article at <https://doi.org/10.1007/s41095-018-0105-0>.

References

- [1] Chen, M.; Mountford, S. J.; Sellen, A. A study in interactive 3-D rotation using 2-D control devices. In: Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques, 121–129, 1988.
- [2] Henriksen, K.; Sporning, J.; Hornbæk, K. Virtual trackballs revisited. *IEEE Transactions on Visualization and Computer Graphics* Vol. 10, No. 2, 206–216, 2004.
- [3] Bier, E. A. Skitters and jacks: Interactive 3D positioning tools. In: Proceedings of the 1986 Workshop on Interactive 3D graphics, 183–196, 1987.
- [4] Conner, B. D.; Snibbe, S. S.; Herndon, K. P.; Robbins, D. C.; Zeleznik, R. C.; van Dam, A. Three-dimensional widgets. In: Proceedings of the Symposium on Interactive 3D Graphics, 183–188, 1992.
- [5] Ware, C.; Jessome, D. R. Using the bat: A six dimensional mouse for object placement. *IEEE Computer Graphics and Applications* Vol. 8, No. 6, 65–70, 1988.
- [6] Zimmerman, T. G.; Lanier, J.; Blanchard, C.; Bryson, S.; Harvill, Y. A hand gesture interface device. In: Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface, 189–192, 1986.

- [7] Froehlich, B.; Hochstrate, J.; Skuk, V.; Huckauf, A. The GlobeFish and the GlobeMouse: Two new six degree of freedom input devices for graphics applications. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 191–199, 2006.
- [8] Perelman, G.; Serrano, M.; Raynal, M.; Picard, C.; Derras, M.; Dubois, E. The roly-poly mouse: Designing a rolling input device unifying 2D and 3D interaction. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, 327–336, 2015.
- [9] Wang, R. Y.; Popović, J. Real-time hand-tracking with a color glove. *ACM Transactions on Graphics* Vol. 28, No. 3, Article No. 63, 2009.
- [10] Wang, R.; Paris, S.; Popović, J. 6D hands: Markerless hand-tracking for computer aided design. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 549–558, 2011.
- [11] Song, P.; Goh, W. B.; Hutama, W.; Fu, C.-W.; Liu, X. A handle bar metaphor for virtual object manipulation with mid-air interaction. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1297–1306, 2012.
- [12] Hancock, M.; Carpendale, S.; Cockburn, A. Shallow-depth 3d interaction: Design and evaluation of one-, two- and three-touch techniques. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1147–1156, 2007.
- [13] Hancock, M.; ten Cate, T.; Carpendale, S. Sticky tools: Full 6DOF force-based interaction for multi-touch tables. In: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, 133–140, 2009.
- [14] Reisman, J. L.; Davidson, P. L.; Han, J. Y. A screen-space formulation for 2D and 3D direct manipulation. In: Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology, 69–78, 2009.
- [15] Martinet, A.; Casiez, G.; Grisoni, L. The effect of DOF separation in 3D manipulation tasks with multi-touch displays. In: Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology, 111–118, 2010.
- [16] Martinet, A.; Casiez, G.; Grisoni, L. Integrality and separability of multitouch interaction techniques in 3D manipulation tasks. *IEEE Transactions on Visualization and Computer Graphics* Vol. 18, No. 3, 369–380, 2012.
- [17] Liu, J.; Au, O. K.-C.; Fu, H.; Tai, C.-L. Two-finger gestures for 6DOF manipulation of 3D objects. *Computer Graphics Forum* Vol. 31, No. 7, 2047–2055, 2012.
- [18] Au, O. K.-C.; Tai, C.-L.; Fu, H. Multitouch gestures for constrained transformation of 3D objects. *Computer Graphics Forum* Vol. 31, No. 2pt3, 651–660, 2012.
- [19] Hachet, M.; Bossavit, B.; Cohé, A.; de la Rivière, J.-B. Toucheo: Multitouch and stereo combined in a seamless workspace. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 587–592, 2011.
- [20] Schmidt, R.; Singh, K.; Balakrishnan, R. Sketching and composing widgets for 3D manipulation. *Computer Graphics Forum* Vol. 27, No. 2, 301–310, 2008.
- [21] Cohé, A.; Décle, F.; Hachet, M. tBox: A 3d transformation widget designed for touch-screens. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 3005–3008, 2011.
- [22] Kin, K.; Miller, T.; Bollensdorff, B.; DeRose, T.; Hartmann, B.; Agrawala, M. Eden: A professional multitouch tool for constructing virtual organic environments. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1343–1352, 2011.
- [23] Sun, Q.; Lin, J.; Fu, C.-W.; Kaijima, S.; He, Y. A multi-touch interface for fast architectural sketching and massing. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 247–256, 2013.
- [24] Paczkowski, P.; Dorsey, J.; Rushmeier, H.; Kim, M. H. Paper3D: Bringing casual 3D modeling to a multi-touch interface. In: Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology, 23–32, 2014.
- [25] Chen, C.-Y.; Yao, C.-Y.; Lin, C.-Y.; Hung, S.-H. Real-time remote control of artificial cilia actuation using fingertip drawing for efficient micromixing. *Journal of Laboratory Automation* Vol. 19, No. 5, 492–497, 2014.
- [26] Hilliges, O.; Kim, D.; Izadi, S.; Weiss, M.; Wilson, A. HoloDesk: Direct 3d interactions with a situated see-through display. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2421–2430, 2012.
- [27] Nancel, M.; Wagner, J.; Pietriga, E.; Chapuis, O.; Mackay, W. Mid-air pan-and-zoom on wall-sized displays. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 177–186, 2011.
- [28] Microsoft. Kinect for Xbox360. 2010. Available at <https://www.xbox.com/enUS/kinect>.
- [29] Chapuis, O.; Bezerianos, A.; Frantziskakis, S. Smarties: An input system for wall display development. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2763–2772, 2014.

- [30] Song, J.; Sörös, G.; Pece, F.; Fanello, S. R.; Izadi, S.; Keskin, C.; Hilliges, O. In-air gestures around unmodified mobile devices. In: Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology, 319–329, 2014.
- [31] Boring, S.; Gehring, S.; Wiethoff, A.; Blöckner, A. M.; Schöning, J.; Butz, A. Multi-user interaction on media facades through live video on mobile devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2721–2724, 2011.
- [32] Boring, S.; Baur, D.; Butz, A.; Gustafson, S.; Baudisch, P. Touch projector: Mobile interaction through video. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2287–2296, 2010.
- [33] Chang, Y.-C.; Wang, H.-C.; Chu, H.-K.; Lin, S.-T.; Wang, S.-P. AlphaRead: Support unambiguous referencing in remote collaboration with readable object annotation. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work and Social Computing, 2246–2259, 2017.
- [34] Kashiwakuma, J.; Kitahara, I.; Kameda, Y.; Ohta, Y. A virtual camera controlling method using multi-touch gestures for capturing free-viewpoint video. In: Proceedings of the 11th European Conference on Interactive TV and Video, 67–74, 2013.



Po-Huan Tseng received his master degree from “National Tsing Hua University”, Taiwan, China. He is currently working at Synology as a software engineer. His research interests include computer graphics and human–computer interaction.



Shih-Hsuan Hung is a research assistant in the GAME center in “National Taiwan University of Science and Technology” (NTUST), Taipei, Taiwan, China. He received his M.S. degree in computer science and information engineering (CSIE) from NTUST in 2016. His research interests are in computer graphics and multimedia.

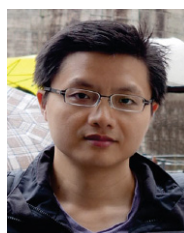


human–computer interaction.

Pei-Ying Chiang is an assistant professor in CSIE at “National Taipei University of Technology”, Taipei, Taiwan, China. She received her M.S. and Ph.D. degrees in computer science from University of Southern California in 2006 and 2011 respectively. Her research interests are in computer graphics and



Chih-Yuan Yao is an associate professor in CSIE at NTUST, Taipei, Taiwan, China. He received his M.S. and Ph.D. degrees in CSIE from “National Cheng Kung University”, Tainan, Taiwan, China, in 2003 and 2010 respectively. His research interest are in computer graphics and multimedia.



focus on shape understanding, smart manipulation, perception-based rendering, recreational graphics, and human–computer interaction.

Hung-Kuo Chu received his Ph.D. degree from the Department of Computer Science and Information Engineering, “National Cheng Kung University”, in 2010. He is currently an associate professor in the Department of Computer Science, “National Tsing Hua University”.

His research interests

Open Access The articles published in this journal are distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Other papers from this open access journal are available free of charge from <http://www.springer.com/journal/41095>. To submit a manuscript, please go to <https://www.editorialmanager.com/cvmj>.