Study and Evaluation of Separability Techniques and Occlusion in Multitouch Surfaces

Jessica Palomares, Manuel Loaiza, and Alberto Raposo

TECGRAF - Technical-Scientific Software Development Institute
Pontifical Catholic University of Rio de Janeiro
Rio de Janeiro, Brazil
jessika.palomares@gmail.com,
{manuel,abraposo}@tecgraf.puc-rio.br

Abstract. Multitouch interfaces allow interacting with a virtual object directly, similar to a real object. However, there are several issues to be resolved, such as the accuracy of the manipulation, the occlusion, the separability of the manipulation, etc. Multitouch interfaces allow multiple spatial transformations that can be performed on a virtual object with only a gesture. For example, an object can be rotated, translated and scaled with two fingers with a single gesture. However, some unwanted movements may occur accidentally. Separability techniques appear with the intent to prevent unwanted movements on multitouch surfaces. Occlusion is another problem that occurs in multitouch interfaces. Often the user's hand hides the vision of the object with which he/she interacts; or the user's action on interface hinders the movement when it clicks on a bottom that triggers action. This paper proposes two techniques of separability, aiming to reduce the problems that arise due to excessive freedom of manipulation in multi-touch interfaces, and evaluates the efficiency of these techniques. The techniques developed are not only applicable in simple virtual objects; they are also for WIMP (windows, icons, menus, pointer) objects, aiming to reduce occlusion. A series of tests was performed to evaluate precision, occlusion time for completion of task, and ease of use.

Keywords: Human-Computer Interaction, multitouch interaction, Separability, Occlusion, Spatial Tranformation.

1 Introduction

Touch-sensitive surfaces appeared as a means to provide a more direct and natural human-computer interaction, allowing creating an alternative to mouse-based interfaces. Among the devices using this technology, we find from mobile individual devices to collaborative tabletop surfaces.

Multitouch tabletop surfaces offer many advantages, such as the detection of several simultaneous touch events along the display area, allowing the parallel interaction of more than one user with one or multiple programs. This technology also enables the manipulation of graphic objects in more complex ways than it is possible

M. Kurosu (Ed.): Human-Computer Interaction, Part II, HCII 2014, LNCS 8511, pp. 293–304, 2014.

with the mouse. A single gesture may generate many simultaneous spatial transformations in an object. For example, with two fingers, users may translate, rotate and scale an object at the same time.

However, there is the opposite situation, where the user only wants to make a subset of these actions, and unwished movements accidentally occur, given our imprecision with fingers movements. According to Nacenta et al. [1], it is difficult only to translate and scale an object without rotating it, since the object reacts to small angle variations among the contact points. This problem can be reduced by a separability technique applied in spatial manipulations in multitouch surfaces.

In the present work, we study the characteristics of separability and occlusion for interaction in multitouch tabletop surfaces. We chose two techniques to demonstrate the importance of these problems and to propose solutions for them. The first technique, called “Handles”, explicitly separates the spatial manipulations by means of areas over the object, as described in Nacenta et al. [1]. The second technique is called “Rock & Rails” and proposes the use of a set of gestures that, combined with touches over the object, separate the spatial manipulations, as described by Widgor et al. [2]. In addition to the separability problem, we show in the present work that occlusion also interferes in the correct spatial transformation of an object, especially in the case of WIMP interfaces, still currently used in multitouch applications.

We propose the modification of Handles [1] and Rock & Rails [2] techniques that, together with other separability techniques and the study of occlusion problems, aim at providing support to the reuse of WIMP interfaces in multiuser devices, such as tabletop surfaces.

This paper is organized as follows. The following section presents some related work. Then, in section 3, the techniques proposed in this work are presented. In section 4 we describe the user tests and analyze the results. Finally, section 5 concludes the paper and indicates future work.

2 Related Work

We present related work divided into two subsections approaching, respectively, the concepts of object manipulation and separability in multitouch surfaces, and the occlusion problem in applications implemented for multitouch surfaces.

2.1 2D Objects Manipulation and Separability

Wu et al. [3] created a prototype for furniture organization in a plan called RoomPlaner. This prototype runs in a DiamondTouch [4] tabletop and can be used by two persons at the same time. Kruger et al. [5] developed the Rotate’N Translate (RNT) technique, aiming to seamlessly integrate the rotation and translation of 2D objects.

Hancock et al. [8], investigated several manipulation techniques for 2D objects and proposed the so called “two-point rotation and translation” technique, also known as rotate–scale–translate (RST), or pinch zoom, when associated only with object’s scaling [6]. In this technique, the first contact point is used to move the object, while the second one is used to rotate it. This technique became very popular in multitouch interaction and is the one we are going to use in the present work as the reference technique, which we call here “no restriction technique”.

In the work of Moscovich and Hughes [7], a new approach is presented. The idea is to make a better use of the number of contact points that our fingers may offer, to map these contact points into events that the user may use to manipulate 2D objects. For example, the Sticky Fingers technique [8] proposes that it is possible to scale, rotate and translate an object using two fingers. Ashtiani e Stuerzlinger [9] introduces a transformation technique with up to three touches called XNT. In this technique, the reference point for rotation and scale transformations is the midpoint calculated using the number of contact points over an object.

In the work of Nacenta et al. [1] proposed a set of interaction techniques that allow users to select a subset of degrees of freedom. The proposed techniques reduced the unwanted manipulations without affecting the performance of manipulation. On them was separated control of the degrees of freedom, thus improving the accuracy of movement that users are able to do about the objects. In the work of Widgord et al. [2] proposed a set of gestures (Rock, Rail and Curved Rail) which, in combination with touches, limited degrees of freedom in spatial manipulations of virtual objects 2D.

Ashtiani and Stuerzlinger [9] proposed a technique called XNT and XNT-S. The technique XNT-S is a variation of the technique XNT, where it is proposed to separate the manipulations by the amount of touches on the object: translation with one touch, two touches to scale and with 3 touch rotation, it was calling XNT-S.

2.2 Occlusion

Multitouch tabletop surface provide the users with a large, horizontal and shared interaction area, offering new opportunities to support collaboration, discussion, interpretation and analysis tasks with the information presented on the surface. However, in a device where input (touches) and output exhibition areas are coincident, the user hand or arm may occlude part of the screen [10]. These occlusion problems have been studied in many Works. For instance, Vogel and Baudish [11] presented the Shift technique, with the argument that it is advantageous to keep the interaction point at the local and to exhibit a copy of the occluded area in another area of the screen.

Roudaut et al. [12] introduced the TapTap technique, designed to improve the direct touch precision, based on a temporal multiplexing strategy. Brandl et al. [13] point that multitouch tables are also affected by the user position and the use of the hands. Based on their observations about occlusion in multitouch tabletops, these authors implemented a digital menu for these surfaces. This menu provides an apperture to avoid the hand occlusion over it, and can be adapted to different hand positions.

Many other solutions have been proposed to avoid occlusion in spatial manipulations, such as a tactile cursor [7], remote manipulation [2], and the handles techniques [1].

Currently, operational systems, such as Windows 7 and 8, Mac OS X, and Linux, provide support for multitouch, and have been changing the size and content of their graphical interface elements. For instance, in some programs provided with Windows 7, such as the Paint, we observe that the toolbar is larger than in the previous version, and it has a new organization of the elements. This indicates that there is a strong tendency to support tactile events, as became clearly evident in Windows 8.

Although Windows 8 provides an interface clearly adapted to multitouch devices, the use of a mouse is still required to interact with windows, which are legacy of previous versions of the operational system. In multitouch interfaces, the direct manipulation of virtual objects is susceptible to occlusion, due to the size of users' hands and fingers. Direct touches increase occlusion, as pointed by Potter et al. [14]. In the present work, we studied the occlusion and how it affects transformation operations (rotation, translation, and scaling) in WIMP interfaces, still present in Window 8.

3 Proposed Separability Techniques

The great advantage of direct manipulation is that it has the potential to increase the velocity of complex manipulations, because this kind of manipulation eliminates the necessity of making the transformation operations sequentially [2]. However, there are tasks that require a higher level of precision, and can be hindered by the control of more than one operation at the same time.

Multitouch tables were developed to allow that users work together and interact simultaneously with an application. The use of WIMP interfaces in multitouch tables requires an adaptation of current Windows, they need to rotate, translate and scale, so that users can execute their tasks more easily. In addition, there are problems related to the occlusion of interface elements. For example, one may trigger an unwanted event when touching a button or menu within the window area.

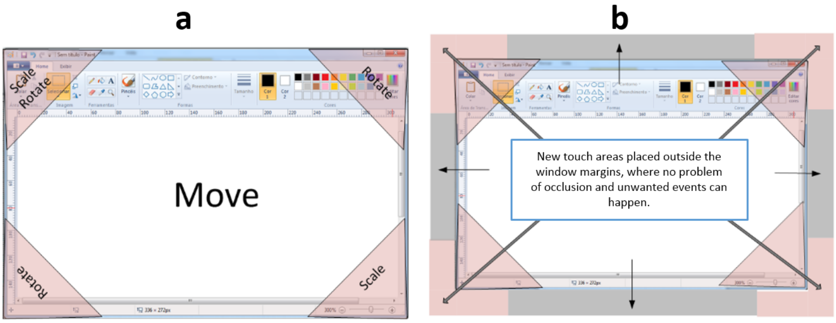
Considering these problems, the present paper proposes two new techniques to support the manipulation of WIMP interfaces in tabletop surfaces. The first technique is based on the work of Nacenta et al. [1] and is called "borders outside the object". The second one is based on the work of Wigdor et al. [2] and is called "with the help of a proxy". These techniques are described below.

3.1 Borders Outside the Object

In mouse-based interfaces, generally the manipulation or transformation of an object requires an explicit way to be executed. For example, in MS Word, if users want to rotate an image, they have to use a specific green manipulator (handle) placed at the top of the image. Handles are in fact a very common approach in traditional interfaces. They provide separability by means of the explicit selection of the transformations that can be applied over an object.

In multitouch interface, handles-based techniques were implemented in the works of Apted et al. [15] and Nacenta et al. [1]. These techniques were proposed as strategies to map and restrict direct spatial manipulation events to specific areas of the objects being manipulated in multitouch surfaces. However, both techniques above may present problems if the user manipulates objects with buttons, menus, or links, present in WIMP interfaces. A single touch over any of these elements may be interpreted as an event associated to the interface element, as a spatial manipulation event.

As an example, consider the case where the user works on a multitouch table and want to show an application window to a colleague, requiring moving and rotating that window. As shown in Figure 1a, using the original Handles techniques, as proposed by Nacenta et al. [1], when rotating the window, the user could accidentally touch window maximization or minimization buttons, or when moving the window, the user can draw something in the drawing area. For these reasons, we redefine the manipulation areas, placing them around the object, and not over it. This change avoids any misinterpretation of the events that could happen due to occlusion or to the size of the user's touch (fat fingers problem [16]).



New touch areas placed outside the window margins, where no problem of occlusion and unwanted events can happen.

Fig. 1. (a) Occlusion problem when applying Handles technique in Windows interface. (b) The first proposed technique: Borders outside the object.

Figure 1b presents our proposal. The semitransparent red areas at the four corners of the object are reserved for rotation and scaling, similar to the original Handles technique. The gray semitransparent areas at the four edges are reserved for the translation. It is important to say that, for rotation and scaling, the user has to touch simultaneously on the two areas corresponding to the transformation. For translation, the user can touch on only one gray area of Figure 1b.

Nacenta et al. [1] pointed some problems related to the areas defined for the handles. One of these problems is that the area defined for each handle is affected by the scaling of the object. If the object is reduced a lot, the user may have difficulties to select a handle. To avoid this problem in our proposal, the borders around the object (our handles area) is maintained with a fixed width, independent of the object size. This size was defined according to the work of Wang and Ren [17], where the authors indicate that interaction targets must have a size larger than 11.52mm, for square objects.

3.2 With the Help of a Proxy

Wigdor et al. [2] presented the Rock & Rails technique for multitouch manipulation based on hand gestures. In this technique, the user makes a gesture with the non-dominant hand and makes direct touches over the object with the dominant hand. Each gesture creates a different kind of object. One of these gestures, called “rock”, creates a semitransparent square called “proxy”, which allows the remote control of more than one object, and also avoids the hands occlusion over the object.

However, the Rock & Rails technique still has a problem in WIMP interfaces. It happens because the user has to keep the non-dominant hand making the selected control gesture over the surface of the object. This may cause unwanted events or occlusion in elements of the application’s interface (Figure 2).

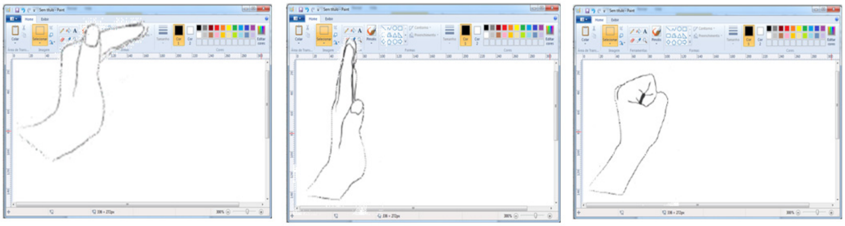


Fig. 2. Hands gesture proposed by Rock & Rails [2] technique and the occlusion problem with WIMP interfaces

In the present work, we decided to restrict the operations to the proxy, since it does not have objects within it and all its area can be used for manipulation. In the proposed technique, we implement the handles technique in the proxy. Therefore, the users may use the proxy when the object connected to it requires a more precise manipulation. The object can also be normally translated with the “no restrictions technique” without affecting the proxy; the user may use the proxy when needed (Figure 3).

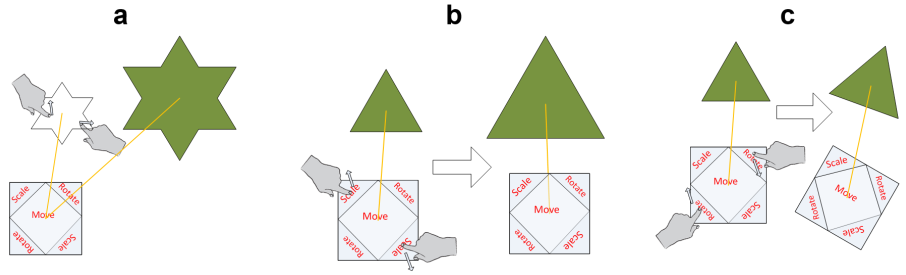


Fig. 3. Proposed technique “with the help of a proxy”. a) Translation can be done using or not the proxy. b) Scale and c) Rotation events only can be done using the proxy.

4 Evaluation of the Proposed Techniques

In this work we are going to compare three manipulation techniques for multitouch surfaces, the two techniques we propose (borders outside the object and with the help of a proxy) and the no restriction technique, which is the “standard technique”, according to Nacenta et al. [1]. To evaluate the techniques, we developed a test application with two scenarios. The first one was developed to evaluate the manipulation of a simple object, in this case, an image (Figure 4a). The second scenario has a WIMP interface to be manipulated, where other events may be triggered during manipulation (Figure 4b).

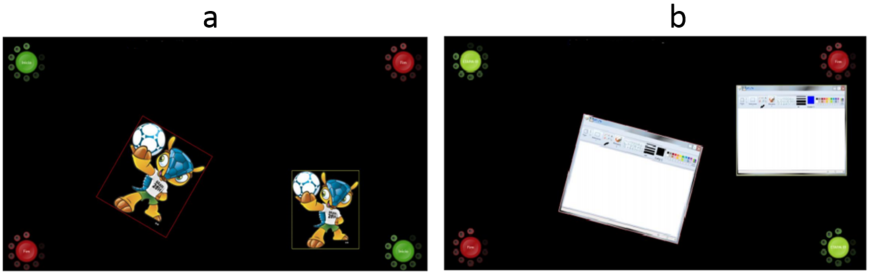


Fig. 4. Screen of scenarios a) an image and b) a WIMP interface

In the second scenario, we developed a window similar to that of the Paint program. In this scenario we want to evaluate the influence of occlusion in the spatial transformations using the proposed techniques. In this Paint-like window, we implemented some of the main characteristics of the program: the color palette (Figure 5a), the pen and eraser buttons (Figure 5b). The drawing area (Figure 5c) and buttons on the upper left corner: minimize, maximize and close (Figure 5d) also trigger events. For example, when one of the buttons is selected by the user touch, the window changes its position to the upper left corner of the screen, showing that an event happened. The application was designed to prevent that this window disappears in case of accidental touches on the close button. The adequate areas for manipulation are highlighted in Figure 5 with red borders.

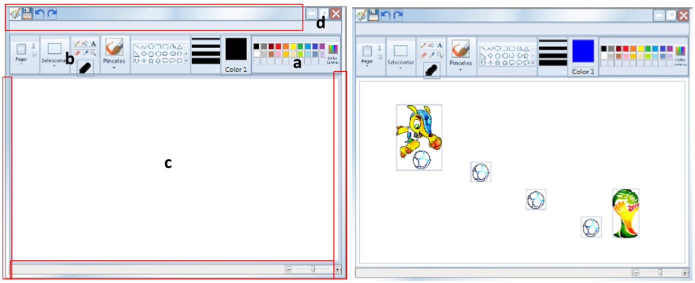


Fig. 5. Paint window style implemented to our test application

In the Rock & Rails technique [2], the user had to use a specific gesture to call the proxy. In our application, the proxy is called when the user performs a tap on the button named "Proxy" showed in Figure 6b. To call the menu, the user must execute the double tap (Figure 6a) anywhere on the table surface.

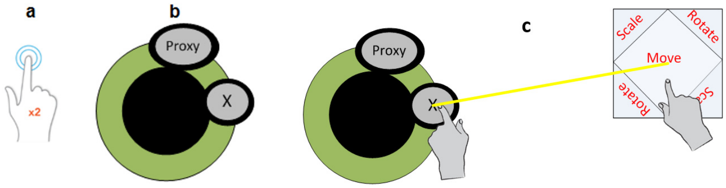


Fig. 6. - a) Double tap gesture, b) Menu interface showed after double tap gesture, c) The proxy and visual connection with the menu interface.

To connect an object to the proxy, the user must make a first tap on the proxy and a second tap on the object, when a line is drawn between the two objects, indicating the connection between them. Then, proxy movements are reflected on the object. The user removes the proxy from the surface through a connection made between the proxy windows and the "X" button on menu content (Figure 6c).

4.1 Test Application

The tests implemented in this work had the objective to evaluate users' performance to complete certain tasks that require spatial transformations on 2D objects. The tests ran on a Microsoft PixelSense SUR40 multitouch table. A group of 20 users participated; evaluating three techniques in two scenarios.

In each scenario the application showed two objects, one object can be manipulated and the other one is static. The user had to align the movable object into another using spatial transformations. In the first scenario, the manipulated object was a simple picture. In the second scenario, the object was a WIMP window. In this second scenario an additional step was defined. After both objects are aligned, the user had to join six images (soccer balls) using dashes. The six images were presented in the Paint window since the beginning of the test (Figure 5). In case the user has drawn something with his/her touches in the drawing area of the Paint window during the window manipulation, he/she would have to first erase these drawings, and then draw the line joining the balls. The purpose of this second step was to reinforce the notion that the manipulation of a WIMP interface might generate unwanted events.

At the end of each scenario, the user had to fill out a questionnaire based on Likert scale where each item had 7 levels of agreement, where 1 means “strongly disagree” and 7, “completely agree”. For data analysis, we define the following variables that allow measuring the performance of each technique: rotation, translation and scale errors, execution time, and time spent by occlusion.

4.2 Analysis of Scenario 1

Figure 7a shows the result of the users' opinion about the ease of use of each technique in spatial transformations like rotation, translation, scaling, and alignment operation in scenario 1. One may observe that the technique with borders outside the object was considered the most difficult, compared to the other two techniques.

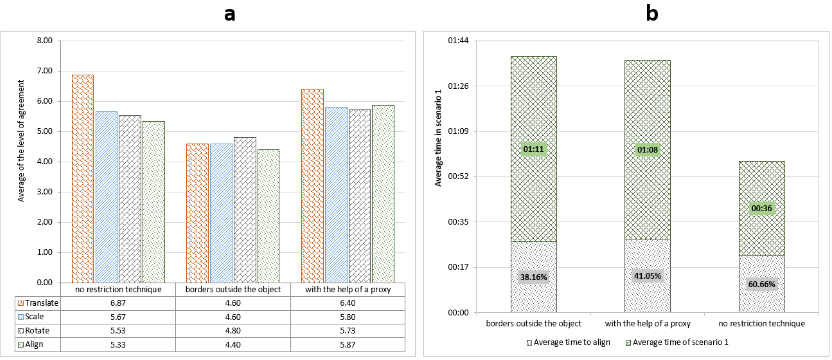


Fig. 7. a) Results of users' opinion considering each technique in translation, rotation, scaling and alignment task. b) Average time to complete the test using the proposed techniques.

Figure 7b shows that the technique with borders outside the objects spent more global time to finish the proposed tasks, compared to the other techniques. However, the figure shows that users spent less time, proportionally to the global time, to do the final alignment task (38.16% of total time). Similarly, despite the no restriction technique presented the lowest mean time to accomplish the task, it was the technique where users spent a larger proportion of time in the alignment task (60.66% of total time).

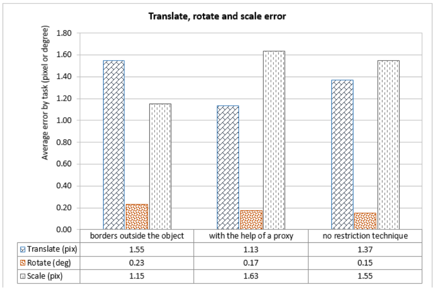


Fig. 8. Average error per task (translation, rotation and scaling)

In Figure 8, we observe the average errors of manipulation tasks. The technique with the help of a proxy presented more error in the scale operation. With respect to translation and rotation, the technique with borders outside the object was the less accurate. These errors measurements reinforce the users' opinion (Figure 8a) with respect to the technique with borders outside the object. Some users said that the view of the object was hampered by the borders when they try to align objects.

4.3 Analysis of Scenario 2

Figure 9a shows the users' opinion about the ease of use of each technique when working with a WIMP window. We observe that in this scenario the no restrictions technique was considered the most difficult. It had the lowest score in all types of spatial transformations. The technique with the help of a proxy obtained users' preference.

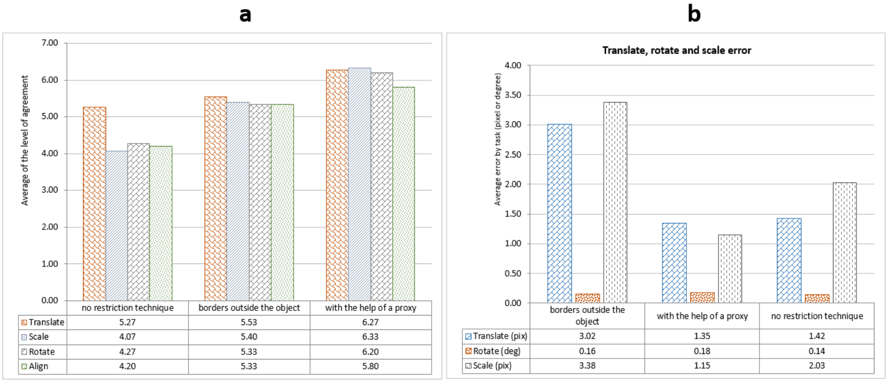


Fig. 9. a) Results of level of agreement of users considering each technique in translation, rotation, scaling and alignment task. b) Average error per task.

We may see a change in users' opinion, related to the use of two fingers to perform rotation and scale transformations. Users now prefer our proposed techniques that provide a wider space for object manipulation, without unexpected events executed by buttons and the drawing area of the Paint-like window interface.

Figure 9b shows the errors with respect to position, scale and rotation of the three techniques. We can see that the technique that got the lowest errors was the technique with the help of a proxy, while the technique with borders outside the object obtained the highest error values for translation and scale.

The second stage of scenario 2 was designed to evaluate how much the occlusion influenced the spatial transformations of the object. In this step, the user was asked to first remove any draft that has been drawn in step one (during spatial manipulation), and then make a trace to join the four balls. The time spent clearing the draft is the measure that we use to indicate the influence of occlusion in each technique. In Figure 10, we observe the average time measured to the occlusion problem. The technique with borders outside the object had the lowest occlusion time and the no restriction technique was the one with a longer duration. We can then infer that this last technique was the most affected by the occlusion.

The technique with borders outside the object was less affected by occlusion because the borders act as a kind of "protection" to prevent users to touch in controls inside the window. The user manipulates only the borders to align the window, for this reason, their exposure to occlusion error was minor compared with the other techniques.

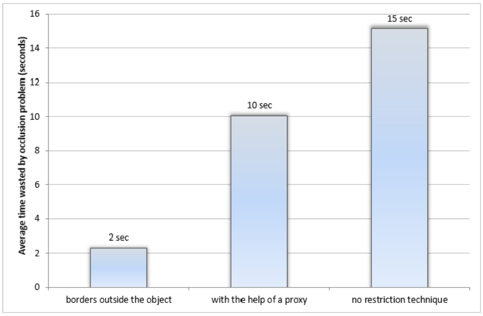


Fig. 10. Average time spent by users to fix problems derived from occlusion

5 Conclusion

In this work, we proposed and evaluated two techniques of separability for virtual objects in 2D multi-touch interfaces. These techniques were evaluated with user tests that found that the two proposed techniques improve separability and reduce occlusion in spatial transformations of simple objects and objects that contain elements of WIMP interfaces.

In relation to the accuracy of each technique, our results suggest that the technique with the help of a proxy improves the separability in both object types evaluated. However, in the evaluation of occlusion interference, it achieved an average result compared to the two other techniques. The technique with borders outside the object has similar gains in separability issue and better performance to reduce the occlusion in objects with WIMP interface. We found that the technique with borders outside the object has a better support when separability and occlusion appear together, especially when we need a better support for reuse of WIMP interfaces.

Our results also indicated the two proposed techniques spend less time in the operation of fine fitting of objects compared with the no restriction technique. Based on these results, we may indicate that the separability is a good strategy for avoiding the time spent in alignment movements and hence the users' fatigue.

Acknowledgments. Alberto Raposo thanks CNPq (process #310607/2013-2) for the support to his research work. This work is also supported by FAPERJ (process #190.243/2013).

References

1. Nacenta, M., Baudish, P., Banko, H., Wilson, A.: Separability of spatial manipulations in multi-touch interfaces, pp. 175–182 (2009)
2. Wigdor, D., Benko, J., Pella, J., Lomabardo, J., Williams, S.: Rock & rails: Extending multi-touch interactions with shape gestures to enable precise spatial manipulations. In: SIGCHI Conference on Human Factors in Computing Systems, pp. 1581–1590 (2011)

3. Wu, M., Balakrishnan, R.: Multi-Finger and Whole Hand Gestural Interaction Techniques for Multi-User Tabletop Displays. In: Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology, pp. 193–202 (2003)
4. Dietz, P.: DiamondTouch: A Multi-User Touch Technology. In: Proceedings of UIST, pp. 219–226 (2001)
5. Kruger, R., Carpendale, S., Scott, S.D., Tang, A.: Fluid integration of rotation and translation. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 601–610 (2005)
6. Wobbrock, J.O., Morris, M.R., Wilson, A.D.: User-defined gestures for surface computing. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, pp. 1083–1092 (2009)
7. Moscovich., T., Hughes, J.F.: Multi-finger Cursor Techniques. In: Proceedings of Graphics Interface, pp. 1–7 (2006)
8. 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, pp. 133–140 (2009)
9. Ashtiani., B., Stuerzlinger, W.: 2D similarity transformations on multi-touch surfaces. In: Proceedings of Graphics Interface, pp. 57–64 (2011)
10. Vogel, D., Casiez, G.: Hand Occlusion on a Multi-Touch Tabletop. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 2307–2316 (2012)
11. Vogel, D., Baudisch, P.: Shift: A technique for operating pen-based interfaces using touch. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 657–666 (2007)
12. Roudaut, A., Huot, S., Lecolinet, E.: TapTap and MagStick: Improving One-Handed Target Acquisition on Small Touch-screens. In: Proceedings of the Working Conference on Advanced Visual Interfaces, pp. 146–153 (2008)
13. Brandl, P., Leitner, J., Seifried, T., Haller, M., Doray, B., To, P.: Occlusion-aware menu design for digital tabletops. In: Proceeding of CHI 2009 Extended Abstracts on Human Factors in Computing Systems, pp. 3223–3228 (2009)
14. Potter, R.L., Weldon, L.J., Shneiderman, B.: Improving the Accuracy of Touchscreens: An Experimental Evaluation of Three Strategies. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 27–32 (1988)
15. Apted, T., Kay, J., Quigley, A.: Tabletop sharing of digital photographs for the elderly. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 781–790 (2006)
16. Wigdor, D., Williams, S., Cronin, M., Levy, R., White, K., Mazeev, M., Benko, H.: Ripples: Utilizing per-contact visualizations to improve user interaction with touch displays. In: Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology, pp. 3–12 (2009)
17. Wang, F., Ren, X.: Empirical Evaluation for Finger Input Properties in Multi-touch Interaction. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1063–1072 (2009)