

Carbon Copy Metaphor: Combining Absolute and Relative Coordinates Inputs for Trackpad

Kaori Ikematsu

Ochanomizu University, Tokyo, Japan
ikematsu.kaori@is.oacha.ac.jp

Siio Itiro

Ochanomizu University, Tokyo, Japan
siio@acm.org

ABSTRACT

This paper proposes a novel input technique involving seamless switching between relative and absolute coordinate modes based on a “carbon copy” metaphor for trackpads. In the method, a small workspace (“carbon copy area”) that corresponds in a one-to-one manner to a trackpad surface is displayed on a computer screen. While working on this virtual carbon copy paper, a user can operate in absolute coordinates; outside the paper, relative coordinates are used to move it anywhere and resize it. Therefore, our technique allows for the appropriate use of relative and absolute coordinate modes with arbitrary timing. This paper discusses the design of the technique, implementation of prototype applications, and a user study in which the technique received positive feedback and was shown to achieve a significantly higher input speed than that of a conventional pointing method.

CCS CONCEPTS

• **Human-centered computing** → **Pointing**

KEYWORDS

Touch/Pointing/Gesture; User Interface Design;

ACM Reference format:

K.Ikematsu and I.Siio, 2017, Carbon Copy Metaphor: Combining Absolute and Relative Coordinates Inputs for Trackpad. In Proceedings of the 29th Australian Conference on Human-Computer Interaction, Brisbane, QLD, Australia, November 2017 (OzCHI 2017), 5 pages. <https://doi.org/10.1145/3152771.3156164>

1 INTRODUCTION

The widespread use of touch surfaces such as in smartphones, tablets, and trackpads implies that touch inputs are commonly observed in everyday life. Currently, two coordinate systems are used in touch surfaces, namely, relative and absolute coordinates. Relative coordinates transform the relationship between the locations of an input device and a cursor. Similar to a conventional mouse, a trackpad uses relative coordinates. When a user moves a finger on a trackpad’s input surface, a cursor in the display follows the finger movement. By changing the control-display gain, even small finger motions can be used to produce large cursor movements.

In this manner, a user can operate over a large screen area even if the input surface is small. Conversely, precise cursor operations can also be performed [6]. However, the relative coordinate method generally requires clutching operations that involve repeated lifting of the finger at the edge of the trackpad for repositioning. Clutching is known to degrade operational performance [4, 13]. In absolute coordinate systems, which are used in smartphones and tablets, the coordinates of the input surface correspond directly to the coordinates of the display. This enables a user to operate graphic user interface (GUI) components directly. Jotting down notes, online signing or drawing a painting is considerably easier when using an absolute coordinate input method because it involves finger motion similar to the pen tip motion used in writing characters or drawing a picture on paper. From the viewpoint of pointing operability, in a sufficiently small input device, the absolute mode is useful for long distance cursor movement [13]; however, it is difficult to operate when the output display size is extremely large.

Both coordinate systems have advantages and disadvantages. Depending on the task or on factors such as screen size, cursor speed, and control-display ratio [3], the appropriate coordinate mode can vary. Thus, higher flexibility might be achieved through the development of a trackpad to enable users to switch between relative and absolute coordinate inputs. In general, trackpads are relative coordinate input based devices and therefore, mapping between a trackpad and workspace to allow for absolute coordinate input to a screen is not trivial. Typically, trackpads and computer screens differ in size. Even if a computer screen corresponds in a one-to-one manner with trackpad dimensions, precise operation is difficult owing to the size differential between the screen and the trackpad, particularly in a large-screen environment [1]. On the contrary, if only a part of a computer screen corresponds in a one-to-one manner with a trackpad, the workspace will be limited. By providing a limited rectangular area on the screen that corresponds directly to a trackpad and enabling a user to input absolute coordinates only in this area and to move the area to any location on the overall screen, the compatibility problem can be resolved. However, an appropriate user interface for moving the area and switching between relative and absolute coordinate modes is required. In this paper, we conducted a user study to investigate an appropriate user interface for three easily switchable modes: (a) a conventional pointer mode with relative coordinates; (b) an absolute coordinate mode for input into a rectangular sub-area on the screen; (c) a mode in which the sub-area is moved. Based on the results of the user study, we

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

developed a novel mode switching technique for trackpads, which we evaluated in terms of efficiency and limitations.

2 Related Work

In this section, we discuss related studies on the development of mode switching methods for absolute/relative coordinate systems and direct/indirect inputs. Fig.1 shows the classification of mode switching methods. In this scheme, direct input is defined as the case in which the input device and output screen are the same (e.g., smartphones). Indirect input is defined as the case in which the input device and output screen are separated (e.g., trackpads/mice).

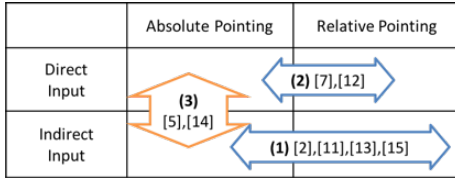


Figure 1. Classification of mode switching methods.

Some related studies on the development of mode switching methods with absolute and relative pointing for indirect input devices such as mice or trackpads are referenced in Fig.1 (1). ARC-Pad [13] provides an absolute pointing mode through finger tapping on a smartphone screen and provides a relative input mode through finger dragging. In this technique, the entire screen corresponds in a one-to-one manner with a smartphone input range, and neither the scaling effect nor the aspect ratio is considered. In [15], which uses a motion-tracked glove, a hand-pointing gesture is used for an absolute-ray casting mode, while opening the hand activates the relative input mode. In [2], absolute indirect mappings are used to control multiple cursors. Inklet [11] is a utility that allows for the use of a common trackpad to emulate a virtual absolute coordinate input area by employing predefined shortcut keys. We propose a technique similar to the above approaches in that it uses explicit mode switching interfaces between absolute and relative coordinates under the control of an indirect input device, i.e., a trackpad. However, previous approaches [11, 15] addressed specific use cases in which a user is locked into a dedicated mode and were not intended to allow for frequent switching between the two modes. We assume that these techniques would enable users to adjust the control- display ratio for absolute coordinate pointing by scaling the workspace to allow for appropriate input.

Some of the related studies shown in Fig.1 (2), [7, 12] involve enabling access to distant and close targets on a large display through the use of a pen device [7] or through bimanual multi touch gestures [12]. The goal of these applications is to achieve efficient long-distance cursor movement or area selection. In contrast, we develop an approach that is suitable for an indirect device with the goal of allowing for efficient cursor positioning in various scenarios.

Several studies have proposed methods for mode switching between direct and indirect input in absolute coordinate situations

(Fig.1 (3)). In [14], a technique for gaze-based mode switching between direct and indirect touch inputs for interactive workspaces combines horizontal and vertical touch surfaces. In [5], absolute mapping at two levels of precision is used for representation. This method provides a constrained area of high-resolution input and a broader area of lower input resolution and is accessed using a dual-leg cursor.

3 CARBON COPY METAPHOR

With the goal of providing (a) a conventional pointer mode with relative coordinates, (b) an absolute coordinate mode for input into a rectangular sub-area on a screen, and (c) a mode for moving the sub-area, we investigated the development of an appropriate user interface. As the behavior of moving a small area to an arbitrary screen location and then inputting into it is similar to drawing and copying using a small sheet of carbon paper on a large drawing sheet, we have adopted the “carbon copy” metaphor. In the system, a virtual “carbon paper” appears on the display when the user touches the trackpad with their fingers. In the real world, we can move a piece of carbon paper by slightly pressing it with our fingers, and we can transfer an image on it to another piece of paper by holding and fixing it. The proposed method is based on this type of hand motion: the user can manipulate the virtual “carbon paper” using actions similar to what they would use to manipulate a physical piece of carbon paper.

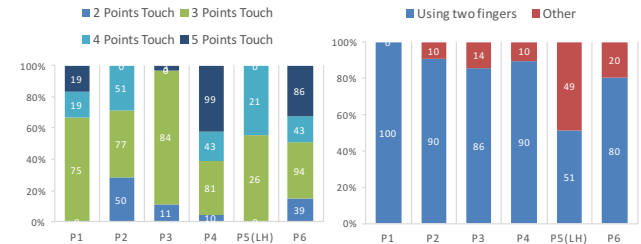


Figure 2. (Left) Frequency of the number of fingers used to hold the carbon paper (%). (Right) Proportion of holding the paper between any two fingertips relative to all methods for holding the paper (%).

3.1 Preliminary Experiments

To design the carbon copy user actions described above, we conducted an observational study on the manipulation of a piece of carbon paper in the real world. We assigned a drawing task to six participants (five right-handed, one left-handed, aged 21 to 24) using a large piece of drawing paper (500 × 400 mm), a small piece of carbon paper (130 × 130 mm), and a ballpoint pen that was designed to run out of ink. Participants signed their names on the left half of the drawing paper and painted freely on the right half. We recorded the experiment and searched for patterns in which the fingers were used for handling the carbon paper.

When the participants wrote or drew on the paper, they tended to press the edge of the carbon paper with their non-dominant hand. This indicates that the participants utilized a wide drawing area and pressed the carbon paper to prevent it from floating or

peeling off. Fig.2 (left) shows the proportion of the time during which fingers touched the carbon paper compared to that required for the execution of the overall task. There are significant variations in the frequencies of the use of various pressing techniques; for instance, using two pairs of three fingers versus other techniques. Variance analysis calculations revealed significant differences in the frequencies of various finger configurations ($F(1,5) = 8.632, p < 0.001$), with multiple comparisons revealing significant differences between the frequency of three- and two-finger touching ($t(5) = 4.651, p < 0.001$), three- and four-finger touching ($t(5) = 3.477, p < 0.005$), and three- and five-finger touching ($t(5) = 4.002, p < 0.001$). Other potential pairings were not considered significant. As shown in Fig.2 (right), we found that the participants tended to widely open two fingers, which implies that they used the two fingers to firmly hold down the carbon paper. Conversely, various finger movements were observed in moving the carbon paper.

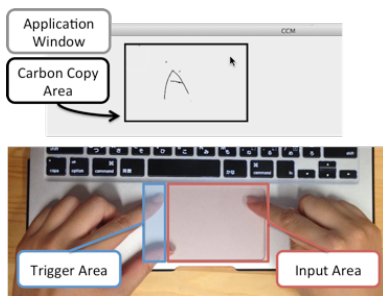


Figure 3. Trackpad divided into two areas (this case is for use by a right-handed person). The black-framed carbon copy area corresponds to the trackpad.

3.2 Interface Design

We designed the interface based on the observational study results. In the interface, two hands can be used to operate the virtual carbon paper. Focusing on participant behavior with respect to writing on the carbon paper, we adopted “pressing the edge of a carbon paper with two fingers of the non-dominant hand spread” as a condition for the writing operation. When users spread two fingers of their non-dominant hand to press the edge of the trackpad while moving the cursor with one finger, a small working area appears at the cursor location on the screen. Gilliot [8] showed that, in absolute indirect-touch situations, the input and output aspect ratios are important, but the display scale is not. Thus, we use the same aspect ratios for the input surface and the screen area. The small working area (carbon copy area) enables an absolute coordinate input mode (carbon copy mode). When users remove the two non-dominant fingers, the carbon copy area disappears and the input mode switches back to the conventional relative coordinate input mode.

4 IMPLEMENTATION

To test the efficiency of the proposed method, we implemented a prototype painting application in Objective-C for use on the OS X

10.9 operating system. In the prototype, touching the trigger area with one finger activates the carbon copy area (the black-framed area on the screen, as shown in Fig.3). Similar to the preliminary experiment, the entire desktop area in the prototype corresponds to the drawing paper, while the black-framed carbon copy area corresponds to the carbon paper. We divide the trackpad into two parts (non-dominant hand side: trigger area; dominant hand side: input area) to define a gesture set for fixing/moving the carbon copy area and switching between relative and absolute coordinate inputs. A mapping between the gesture set and the functions of the application is shown in Fig.4.

No.1 involves one finger moving on the trackpad, corresponding to cursor movement with conventional relative coordinate inputs.

No.2 involves two fingers fixed on the trigger area. Using this operation, the carbon copy area appears (initially at the same size as the trackpad) on the screen.

No.3 involves two fingers fixed on the trigger area and one finger moving on the input area. Using this operation, a user manipulates the pointer inside the carbon copy area in the absolute coordinate input mode. Our prototype produces cursor movement by applying a dragging motion (moving while pressing) to the input area. That is, when users activate the carbon copy mode, they can trace the input area to obtain a preview of cursor movement and they can drag to draw lines on the paint application. The cursor follows the movement of the finger on the input area.

No.4 involves one finger on the trigger area and one finger fixed on the input area. This corresponds to positioning the carbon copy area in relative coordinates.

No.5 involves a scaling function for the carbon copy area. It can be activated by fixing two fingers on the corner of the trigger area and then scaling the carbon copy area (while maintaining the aspect ratio) by moving one finger along the input area. In this method, finger motion away from the trigger area enlarges the carbon copy area while motion toward it shrinks the workspace.

Using the above functionalities, the proposed method switches between the respective coordinate modes by simply mimicking the manner in which writing is performed on carbon copy paper.

No.	1	2	3	4	5	6*
Fingertip position on a trackpad						
Functional Instruction	Moving a Cursor Pointer (Relative Coordinates)	Showing a Carbon Copy Area	Moving a Cursor Pointer (Absolute Coordinates)	Moving a Carbon Copy Area (Relative Coordinates)	Scaling a Carbon Copy Area	Scaling a Carbon Copy Area to fit a forefront window

6* : only available in the GUI application.

Figure 4. Mapping between touch gestures and functions. Red circles indicate a moving finger and blue indicate a fixed one.

5 USER STUDY

5.1 Experimental Setup

To measure the effectiveness of the proposed method, we compared the following two methods: Method 1, i.e., the

proposed technique (combining relative and absolute coordinate inputs), and Method 2, i.e., employing a conventional relative coordinate input using a trackpad. We tested two main hypotheses: (1) after a short learning time, the performance of Method 1 gradually increases; (2) Method 1 provides better performance than Method 2. For comparison, we developed a simple paint application on Apple MacBook Pro (13 inch, trackpad: 105 × 75 mm). Eight people participated (six right-handed, two left-handed, aged 22 to 65) in the experiment. All but one of the participants used an Apple Magic Trackpad on a daily basis. Each participant was asked to write animal names (each having five stroke counts) in English within a series of blue boxes (60 × 43 mm) (Fig.5 (left)). We set the carbon copy area to 70 × 50 mm, giving it and the blue boxes the same aspect ratio as the trackpad embedded in MacBook Pro. Each task consisted of writing five animal names, and each participant was asked to perform ten such tasks for each technique. Thus, we obtained 2,000 strokes (8 [participants] × 10 [task/participants] × 5 [words/task] × 5 [strokes/word] = 2,000). Before conducting the tasks, we demonstrated them using each technique. Participants were provided a maximum of 5 min to practice performing each technique, with no instructions given concerning the order of writing words. In the proposed technique (Method 1), the carbon copy area can be displayed and positioned through relative coordinate inputs, and then, a participant can write words using absolute coordinate inputs. Each participant employed both methods in a counterbalanced order. We observed the changes in the performances of the respective techniques through learning and practice. We recorded task completion time and error counts, i.e., the number of unintentional mode switches for each trial, and each task was video recorded in its entirety.

5.2 Result

Figure 5 (right) shows the input speed results for each technique. Initially, Method 1 achieved a lower average input speed (70 s) than Method 2 (50 s). However, participants rapidly learned how to use our technique and began to achieve faster input speeds than those under Method 2 starting from the fifth writing task trial. As the comparison experiment continued, our technique continued to achieve faster input speeds. By the end of the study, the average input speed of Method 1 was 29 s, while that of Method 2 was 43 s. The increase in performance observed in the study occurred through short-term learning and practice, supporting our hypothesis (1). The execution times of the final trials using Method 1 were considerably shorter than that of the initial trial (the latter trials were shorter by 41 s on average). In contrast, the average difference for Method 2 was only 7 s. Focusing on the tenth trial of the writing task, a t-test revealed that the difference between Methods 1 and 2 was significant ($t(7) = 4.10122, p < 0.01$). We believe that this is attributable to the fact that all participants regularly use relative coordinate input devices (mouse or trackpad). The participants familiarized themselves with our technique after only a few trials, and as a result, they could operate the application more efficiently and reduce their execution time. The results of the experiment demonstrate that after a short

learning period, the proposed technique becomes more efficient than a conventional input technique, thus supporting hypothesis (2). In the Method 1 trials, all participants utilized the absolute coordinate mode for implementing the writing action and the relative coordinate mode for moving the cursor in the intervals between writing each animal name. We presume that the participants intuited that the relative coordinate mode is useful for short distance cursor movement and the absolute coordinate mode is suitable for writing. In terms of error counts, 20 unintentional mode switching occurred. However, the opposite case never occurred. The total error rate was $(20 \times 100) / 2000 = 1.00\%$.

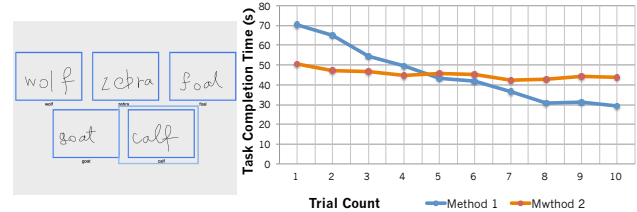


Figure 5. (Left) Five animal names in lowercase, produced using five strokes for each word. (Right) Mean input speed.

6 DISCUSSION AND LIMITATIONS

In the use of the non-dominant hand for subsidiary tasks such as temporal mode switching [9, 10], our interaction design follows previous studies. The proposed technique utilizes the non-dominant hand for temporal mode switching modality. We received favorable comments from seven participants to the effect that the mode switching operation based on the carbon copy metaphor is easy to understand. The metaphor of “while pressing the edge of the carbon paper, the paper is fixed and transfers along the locus of a pen tip” is intuitive and reasonable. In addition, four participants mentioned that switching only while touching the trigger area is similar to a modifier key function, indicating the usefulness of the proposed technique, particularly for short duration tasks involving large cursor movements (from one screen to another screen in a dual display environment or from a corner to an opposite corner), online signing, jotting down notes, etc.

6 CONCLUSION

In this paper, we proposed a novel trackpad-based mode switching technique developed using data obtained from an observational study of real-world user manipulation of carbon paper. The proposed technique achieved a significantly higher input speed than that of a conventional input method and received positive feedback from participants. Future work will involve further testing as to how our approach works at other screen sizes.

ACKNOWLEDGMENTS

This work was partially supported by JSPS KAKENHI Grant Number 26330219 and 17J02834.

REFERENCES

- [1] Accot, J., and Zhai, S. Scale effects in steering law tasks. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '01, ACM (New York, NY, USA, 2001), 1–8.
- [2] Benko, H., Izadi, S., Wilson, A. D., Cao, X., Rosenfeld, D., and Hinckley, K. Design and evaluation of interaction models for multi-touch mice. In Proceedings of Graphics Interface 2010, GI '10, Canadian Information Processing Society (Toronto, Ont., Canada, Canada, 2010), 253–260.
- [3] Benko, H., Wilson, A. D., and Baudisch, P. Precise selection techniques for multi-touch screens. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '06, ACM (New York, NY, USA, 2006), 1263–1272.
- [4] Casiez, G., Vogel, D., Pan, Q., and Chaillou, C. Rubberedge: Reducing clutching by combining position and rate control with elastic feedback. In Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, UIST '07, ACM (New York, NY, USA, 2007), 129–138.
- [5] Debarba, H., Nedel, L. P., and Maciel, A. Lop-cursor: Fast and precise interaction with tiled displays using one hand and levels of precision. In 3DUI, M. Billinghurst, J. J. L. Jr., and A. Lcuyer, Eds., IEEE (2012), 125–132.
- [6] Douglas, S. A., and Mithal, A. K. The effect of reducing homing time on the speed of a finger-controlled isometric pointing device. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '94, ACM (New York, NY, USA, 1994), 411–416.
- [7] Fortlines, C., Vogel, D., and Balakrishnan, R. Hybridpointing: Fluid switching between absolute and relative pointing with a direct input device. In Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology, UIST '06, ACM (New York, NY, USA, 2006), 211–220.
- [8] Gilliot, J., Casiez, G., and Roussel, N. Impact of form factors and input conditions on absolute indirect-touch pointing tasks. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14, ACM (New York, NY, USA, 2014), 723–732.
- [9] Guiard, Y. Asymmetric division of labor in human skilled bimanual action: the kinematic chain as a model. *Journal of Motor Behavior* 19, 3 (Sept. 1987), 486–517.
- [10] Hinckley, K., Pausch, R., Proffitt, D., Patten, J., and Kassell, N. Cooperative bimanual action. In Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '97, ACM (New York, NY, USA, 1997), 27–34.
- [11] Inklet. <https://tenonedesign.com/inklet.php>.
- [12] Kuribara, T., Yoshikawa, T., Shizuki, B., and Tanaka, J. Handyscope: A remote control technique using pull-out gesture. In CHI '14 Extended Abstracts on Human Factors in Computing Systems, CHI EA '14, ACM (New York, NY, USA, 2014), 175–176.
- [13] McCallum, D. C., and Irani, P. Arc-pad: Absolute+relative cursor positioning for large displays with a mobile touchscreen. In Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology, UIST '09, ACM (New York, NY, USA, 2009), 153–156.
- [14] Voelker, S., Matvienko, A., Schoening, J., and Borchers, J. Combining direct and indirect touch input for interactive workspaces using gaze input. In Proceedings of the 3rd ACM Symposium on Spatial User Interaction, SUI '15, ACM (New York, NY, USA, 2015), 79–88.
- [15] Vogel, D., and Balakrishnan, R. Distant freehand pointing and clicking on very large, high resolution displays. In Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, UIST '05, ACM (New York, NY, USA, 2005), 33–42.