BezelCopy: An Efficient Cross-Application Copy-Paste Technique for Touchscreen Smartphones

Chen Chen Simon T. Perrault Shengdong Zhao Wei Tsang Ooi

Department of Computer Science, School of Computing, National University of Singapore
13, Computing Drive
Singapore, 117417
{chchencc, perrault.simon}@gmail.com, {zhaosd, ooiwt}@comp.nus.edu.sg

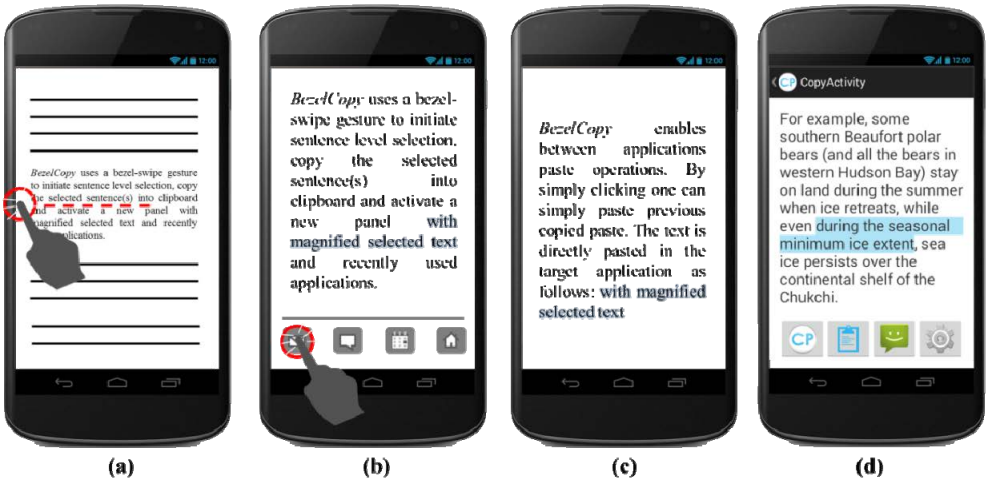

Figure 1: Step-by-step description of copy paste operation with BezelCopy (a) User performs a bezel gesture to select the desired sentence(s); (b) the magnified selected text appears on a new panel to enable fast and precise selection; a list of application icons appears on the bottom of the screen which can be selected as the target application to paste the text; (c) the text is appended to the end of the document in target application. (d) actual screenshot of the panel (b) implemented on a Nexus S running Android 4.1.2.

ABSTRACT

Copy-Paste (CP) operations on touchscreen smartphones are not as easy to perform as compared with similar operations on desktop computers. The smaller screen size and input area make both text selection and application switching more difficult to perform. To enable faster copy-paste on touchscreen smartphones, we introduce *BezelCopy*, a copy-paste technique that uses a bezel-swipe gesture to determine a rough area of interest in the document. Chosen text is magnified in a new panel to enable fast and precise selection. With the new panel, users can perform easy tap-and-drag gestures to select the exact content, and tap the application icon on the bottom of the panel to paste it to the target application. Users can further adjust the location of the pasted text in the target application using drag and drop. We conducted two experiments to compare the performance of *BezelCopy* with alternative approaches, and our results show that *BezelCopy* outperform existing copy-paste techniques for a number of commonly performed copy-paste tasks.

Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces

AVI '14, May 27 - 30, 2014, Como, Italy

http://dx.doi.org/10.1145/2598153.2598162

General Terms

Human Factors

Keywords

Mobile computing; handheld device; copy-paste; bezel interaction; input; smartphone

1. INTRODUCTION

Copy-paste (abbreviated as **CP** in rest of the paper) is a ubiquitous computing operation that is carried out daily by most computer users [20]. It is generally considered as a simple operation on the desktop computer (although situation can be slightly more complex for cross-document copy-pastes) [20]. However, the emergence of touchscreen smartphones has largely changed this perception.

Smartphones have enjoyed high growth in recent years with approximately 950 million of them sold in 2013[1]. For many users, smartphone has become the primary (or only) computing device they use. Without a convenient and easy to use copy and paste operation, it can cause delays in productivity and frustration among users. To perform a CP operation, users usually go through four main steps: initiate and adjust selection, issue copy command, switch application, activate paste button and paste. Among these steps, selecting source text and switching between applications are the two more tedious components of the CP

[1] http://www.gfk.com/news-and-events/press-room/press-releases/pages/demand-for-mobile-phones-at-record-high.aspx

operation on smartphones according to our preliminary study with 17 users. Selecting text in touchscreen smartphones is more difficult than in desktop, since the finger is not as precise as the mouse [5]. Users often need to spend more efforts to adjust the caret to fit the granularity of the source text, especially when the font size is small. Some measures have been taken for selecting the text more precisely: on iOS, the magnified excerpt of the text will be shown when moving the caret. This inconvenience is further magnified with the limitation of the small screen, which requires more efforts to scroll or to switch applications.

In this paper, we aim to improve the CP operation on touchscreen smartphones. We first performed a step-by-step analysis of the existing copy-paste approaches in today's mobile phones, identifying the sources of difficulties. We then designed and implemented a novel copy-paste technique called BezelCopy. We evaluated BezelCopy with two existing techniques on Android phones. We found that BezelCopy can significantly improve the copy and paste performance as compared to the two alternative approaches. We also compared our method with BezelSwipe [14] and found that BezelCopy, while requiring the use of an additional panel, is still faster for CP operations. We concluded with a discussion of potential applicability of the lessons learned in our iterative design and study process to help improve the design of other interaction techniques for touchscreen handheld devices.

2. WORKFLOW ANALYSIS OF EXISTING SMARTPHONE CP TECHNIQUES

Existing CP techniques implemented in popular smartphone operating systems more or less follow a similar workflow, which consists of the following steps:

Initiate selection. This step is typically achieved by either a long press (iOS and Android, Nokia N8, Blackberry Torch) or a tap (Windows Mobile 7). Selection mode is initiated with the text under the finger highlighted.

Adjust selection. The highlighted region comes with two adjustment sliders at each end of the selected text, allowing users to further adjust the highlighted region by moving the two adjustment sliders associated with the starting point and the end point of the selection. Since the text size on the mobile device is typically small, and the precision of finger selection is not very precise, iOS and early Android (before version 3) provide a Magnifier to enlarge the text directly under the finger so that users can perform the selection easier. Apple claims that the content magnifier should include portions of three lines of text [13]. Since version 3, Android phones using a much bigger draggable selection anchors instead of a magnified lens (Figure 2). BlackBerry phones use the "Alt" key and trackball to adjust the selection.

Issue copy command. For all the systems, a copy button typically appears within a delay of 500-1000ms after the long pressing. The users should explicitly press the copy button to copy the selected text after the adjustment of the selection region.

Switch application. To paste the copied text to another application, one needs to switch to that application first. The typical approach is by pressing the home button (iOS, Android). On iPhone, users can press home button to go back to the main page and select the target application, or double click home button to activate the mostly recently used app list. In Android, depending on different versions, users can either press home button, long press home button or use navigate button to switch to target application. In Windows Phones, users can press and hold the back button to see the recently used app. In blackberry phones, users can use "Alt" key and "Esc" key to show recently used apps and use track wheel to navigate to the target application.

Activate Paste Button. We should explicitly activate the paste button before we perform the paste action. The paste button typically appears after a tap (on iPhone) or a long press (on Android) at the location of paste.

Paste. The most common type of paste on mobile phones is slightly different from desktop computers. On desktop computers, before paste, one first needs to navigate the cursor to the location for paste. On mobile phones, it is less common. According to our interview with the participants, most pastes are direct paste to SMS, chat service (Whatsapp, WeChat, etc.), social network service (Facebook, Twitters, etc). In such cases, one just need to select the text field in which the text will be pasted to, and issue the paste command by tapping the "paste" button.

For non-selectable items (e.g., SMS messages, headings of email messages, maps, links in the browser), typically the "copy" and "paste" commands appear after a long press on the item to be copied, without allowing users to enter the selection mode and adjust the handles.

Additional gestures are implemented by various smartphone OS to speed up the selection process. For instance, quadruple taps select the whole paragraph on iOS, and long press with two fingers selects the text between the fingers on Android.

2.1 Built-in Copy-Paste method on Android

The CP methods on smartphones are more or less similar. The iOS implementation of CP technique is more or less the same with Androids, except the Android CP techniques implements much bigger anchors to support easier text selection adjustments. In this paper, we choose the two built-in CP methods on Android platform: *default* method and *2-fingers* method, as baselines for our studies.

Default method. It's the most commonly used method among Android users. This method uses long pressing a single word to initiate the selection, dragging the left and right sliders to adjust the selection, pressing copy button to copy, pressing "home" key to switch app, long pressing to activate paste button and finally pressing the button to complete paste.

2-Fingers method. The difference between 2-fingers method and default method is that the users can use two fingers to initiate selecting a chunk of text. The text between the two fingers will be selected after the 2-fingers' long pressing. The other steps are the same with the default method. The 2-fingers selection method is developed only in the Textview component of Android and iOS.

3. RELATED WORK

BezelCopy takes advantage of bezel interaction to provide a simple and non-ambiguous way to operate CP cross and within applications. We briefly review the related literature to bezel gestures and CP techniques below.

3.1 Bezel Interaction

Bezel gestures [3] are gestures that are started on the bezel of a device and then continue on the touch screen. A left bezel gesture example is shown in Figure 1(a). This kind of gesture is simple to detect and is conflict-free with over pre-defined gestures, such as tapping, panning and zooming, as stated in [[14], [16], [17]].

The bezel itself can also be used as a passive tactile landmark to guide the user's finger, as demonstrated on wristwatches in [1].

Some works explored the potential of bezel interaction in low visual attention situations. In particular, Serrano et al. [16] found

out that it is possible to discriminate up to 5 different zones on the bezel of a 10" tablet even eyes-free. Jain and Balakrishnan [10] demonstrated the ability for users to efficiently type text on small devices by performing bezel gestures from specific areas, with the required visual attention decreasing over time.

BezelSwipe [14] was the first work to introduce bezel gestures and was designed to ease the process of image and text selection to shorten the copy-paste time. This method improves the first two steps as mentioned in previous section. However, BezelSwipe focuses on the non-conflict component of the technique, and still needs improvement of the complete copy-paste process, such as multi selections or between applications paste mechanism.

3.2 CP Techniques

3.2.1 On multi-touch devices

Copy-paste mechanism was implemented quite late on mobile systems: it was first available on iOS 3 (iPhone) and Android 1.5.

Few works has investigated CP on mobile phone specifically. BezelSwipe [14] introduced bezel gesture to make the selection faster, however they didn't dig deeply how to make the whole Copy-Paste operation faster.

Fuccella et al. [7] showed that gestures tend to be significantly faster than graphical widgets for selection and CP operations. They also considered the font size of the text for such operations and showed an effect of the font sizes considered as factors.

SPARSH [12] designed a touch to copy, touch to paste prototype. It used cloud to transfer and restore copied data and used a user-defined unique gesture pattern to identify different users. This method was designed to apply copy-paste between different digital devices, and could also be regarded as a data transfer method. However, this method requires a good network and the gesture may conflict with existing ones.

Memory Stones [8] proposed a "pick up", "carry" and "put down" process to perform copy-paste operation. They required users to maintain their fingers position as they were carrying a "stone" to finally get the "object" which they picked up previously. As SPARSH [12], they restored the "object" picked into a server and they got the "object" by the unique fingertip shape, so they met the same network and gesture confliction problem as SPARSH.

3.2.2 On desktop computers

AutoComPaste [20] integrated auto-completion technique into traditional copy-paste operation. It defined another copy-paste workflow which was "type in prefix", "select from candidates" and "adjust results". This method was significantly faster than traditional ones when the users know the prefix of the text to copy. However, the necessary prefix knowledge made it not a general CP solution.

Citrine [18], Entity Quick Click [2] and smart copy and paste (SCP) [9] explored methods to copy-paste special structure of contents such as address and phone numbers. They aimed to identify the content structures and organized them for pasting.

[4] and [6] introduced Windows management methods for crossing-window copy-paste on desktops. [4] proposed *restack* and *roll* as two new methods to facilitate switching between source and target window. [6] presented *history manager, desk pop* and *stack leafing* and integrated these techniques with traditional drag-drop to fast the copy-paste operation.

Both special structure techniques and Windows management techniques aimed to optimize a single step in CP process, i.e. text selection and application switching. They didn't consider changing the traditional CP workflow.

4. PRELIMINARY STUDY

We conducted a preliminary study on 18 users (6 females, aged 20-25, average: 22.8, who also took part in the first user study). 17 of them owned a smartphone with touchscreen (10 Android, 6 iPhone and 1 Windows Phone) except for a man who owned a regular Nokia phone. Thus, we only performed this preliminary study with 17 participants.

4.1 Objectives and Procedure

The goal of this study was to understand how users usually perform copy-paste operations using existing techniques on their smartphones, whether they would perform this operation with one or two hands. We were also interested in the text granularity of their CP operations, namely if they tend to select words, phrases or complete sentences.

To observe how CP is performed, we asked our participants to complete a series of CP operations using their own mobiles. We asked them to access to a random Wikipedia page, separately copy a word, a phrase, a sentence and a paragraph as they liked to the message app, mail app or notes app. Once they finished the operations, participants had to complete an online questionnaire with close ended questions, asking their frequency of performing CP, applications involved in CP, text granularity in CP and their troubles when performing CP.

4.2 Posture for CP

Among the 17 users, we noticed that the most common posture for CP is using two hands to hold the phone and using the thumb to select (Figure 2): 9/17 users naturally adopted this posture. 3/17 tended to use right hand to hold and thumb to select. Another 3/17 used left hand to carry the device with the index finger to select. Finally, 2 users would just put the phone on the desk.

4.3 Use of CP on smartphone

We asked our participants the kind of applications they would usually copy text from. We found majority of users use CP either while they are surfing the Internet (16/17 participants) or messaging (9/17). The most common applications they would paste the copied content to include Messaging Tools (16/18) followed by e-mails applications (8/17) and notes applications (6/18). The performed CP operations usually involved either groups of words (11/17) or isolated words (7/17).

Overall, 15 participants suggested that at least half of their copy-paste operations were cross-application. This percentage is consistent with results obtained from a previous study on desktop applications [20].

4.4 Difficulties with CP operations

Participants had to rate the overall perceived difficulty of CP operations on their phones on a 5-point Likert Scale. Overall, CP is perceived as a marginally difficult operation (average score 2.6/5). In addition, 9 users found it very difficult or difficult to perform. The mentioned reasons of difficulty included text selection (for copy) is time-consuming (mentioned by 13/17 participants) and also error-prone (8/17). They also found application switching time consuming (7/17).

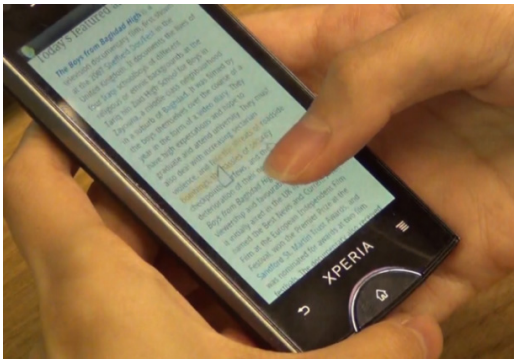

Figure 2: Traditional copy-paste operation on Android. Users tend to use both hands to hold the device, and do the text selection with the thumb finger of the dominant hand.

In summary, the preliminary study revealed that our participants usually perform CP applications mainly with two hands (9/17), and that they tend to copy words or groups of words (11/17) quite often between applications. Finally, the current CP seems usually quite bothersome and difficult to them since they find this operation time-consuming and involving too many steps.

5. BEZEL COPY

Based on the results of our preliminary study, we designed BezelCopy. BezelCopy reduces the time needed to perform CP on the smartphone by removing the most time-consuming reduplicative steps and replace some gestures which are difficult to perform or tedious. BezelCopy (i) introduces a multi-function bezel swipe gesture to reduce the time needed to initiate the selection, (ii) magnifies the selected region in a new panel to make it easier to select by simple tap-and-drag gesture, (iii) displays recently used applications in the new panel to reduce the time to switch to the target application, and (iv) automatically pastes the selection into target application immediately after switching. Details of each step are further elaborated below.

5.1 Interface and Interaction of BezelCopy

The core UI component of BezelCopy is a CP panel, a UI component that shows the candidate text to copy, along with the list of recently used applications. Figure 1 shows a step-by-step description of the technique. Based on the CP panel, the CP process involves 2 to 4 steps depends on different scenarios.

Activate CP panel. To activate the CP panel, we implement the bezel gesture: The user swipes his finger starting from the left vertical screen edge, and ending in the text display areas. The sentence(s) which was/were crossed by the finger will be selected and copied to clipboard at the same time. The CP panel will pop up immediately after the fingertip leaves the screen and the selected sentence(s) will be displayed on the CP panel. This step is partially equivalent to the step of initiating the selection in existing CP techniques except we automatically put the selected content to clipboard.

Adjust selection. Once the CP panel is activated, the system tacitly approves that the next operations are for selecting and copying. So we use a tap, or tap and drag gesture to complete the more specific selection. The system will implicitly copy the selected text to the clipboard at the same time. Thus, we simplified the adjustment process by removing the step of tapping copy menu to explicitly copy. Since the text selected in the first step has already been copied, if the user doesn't want to adjust the content, s/he can skip this step. The CP panel also simplifies the adjustment step by magnifying the sentence(s) selected in the first step to fill the panel which is almost the size of the screen. BezelCopy ensures that the font size of the text in the CP panel display is never smaller than either the text in the source document or 18sp (18sp is the default medium font size to display text on Android OS. If the global text scale is 100%, then one sp is one pixel on a 160 dpi screen[2]), and if rendering of the selected paragraphs is too large to fit into the CP panel, the text will overflow and the user needs to scroll to see the "other" text.

Select target app. Once the adjustment is finished, the user can either close the CP panel to return to the source application, or switch to the target application to paste the text. The CP panel displays a row of recently used applications at the bottom of the panel, which the users can scroll through and tap on the icon of the target application to switch to. By displaying these application lists, we remove the need for the user to explicitly calls up the list of running applications (e.g., by clicking the 'home' button) or go back to home screen, shortening the time to switch applications. Selecting the target app also signifies pasting. After the user switched to the target application, the copied text is automatically pasted into the recently focused text editor in the application.

Adjust pasting. The pasting action is automatically performed with selecting the target app. After the target app is showed up, the pasted text will remain selected so that the user can easily drag the text and relocate it to the correct place if needed, or delete the text, if pasting is not the intention.

Table 1: Comparison of BezelCopy with traditional CP techniques on Android

CP step	BezelCopy	Traditional	2 fingers
Initiate selection	Bezel gesture	Long press	Long press (2 fingers)
Adjust selection	Tap-Drag on magnified text	Sliding finger	Sliding finger
Issue copy command	N.A.	Tap copy button	Tap copy button
Switch application	Tap the target app displayed on CP panel	1. Activate app list or go back to home screen. 2. Select the target app.	1. Activate app list or go back to home screen. 2. Select the target app.
Activate Paste Button	N.A.	Long press	Long press
Paste	N.A.	Tap paste button	Tap paste button
Adjust pasting	Drag-Drop	N.A.	N.A.

In these steps, the first step and the third step are necessary and the other two steps are optional. It means that we can perform a CP operation with only two steps at least compared to the original methods with six steps at least (five steps for word CP). Table 1 compares BezelCopy with the existing CP techniques on smartphones. The main advantages of BezelCopy is that triggering selection with a bezel drag should be quicker than a long press; the magnification screen to properly adjust text selection would make adjusting easier on small fonts; the list of

[2] http://developer.Android.com/design/style/typog raphy.html

applications also eases cross-application CP and finally, the paste is automated when the user switches applications.

6. USER STUDY 1: COMPARISON WITH EXISTING CP METHODS ON ANDROID

6.1 Participants

Eighteen right-handed participants (12 males and 6 females, age ranged from 20 to 25 years, M=22.8, SD=1.68), recruited from within the university community, participated in the user study. All of them have used copy-paste for mobile phone. 17 of them are smartphone users, while 1 has a normal phone.

6.2 Apparatus

The experiment was conducted on a Nexus S, manufactured by Samsung, running on Android 4.1.2 Jelly Bean. The smartphone has a 1 GHz single-core ARM Cortex-A8 processor, with a 4.0 in (100 mm) diagonal Super Clear LCD display with 480x800 px (223 ppi) and 9:15 aspect-ratio. The BezelCopy technique was implemented in Java7 and based on Android SDK API v8.

6.3 Task and Stimuli

All participants were asked to copy text from a source application and paste it into a different application for the three copy-paste techniques. We prepared four different sections from three articles with words ranged from 361 to 435.

In the experiment, we compared the performance of three CP methods: default Android method, 2-fingers selection method and BezelCopy method. In addition to the copy-paste techniques, we also considered two more factors, font size and granularity, which can affect the performance of the copy-paste task.

Technique In this experiment, we compared the default method, 2-fingers method and BezelCopy method, which are all introduced in previous sections.

Font Size As the text size can affect how easy or difficult one can select the text, we included three different font sizes specified in scale-independent pixels (sp) in our study, which were 15sp, 18sp and 21sp. The two adjacent font size 15sp and 21sp were included to verify the effect of font size for CP in this study.

Text Granularity The text to copy can vary in its length, ranging from word, phrase, sentence, to paragraph. To assess the difference, we included text granularity as a control condition.

6.4 Design

A within-participants design was used. Each participant performed copy-paste task using all three techniques. The orderings of the three techniques, the articles and the font size were counterbalanced across participants using Latin square.

At the beginning of the experiment, we gave participants a learning session of 24 trials to help them become familiar with the task. This session consisted of users performing CP for each technique and four granularities. Participants completed the complete experiment in approximately 40 mins, including required breaks and training.

Thus our design (excluding practice trials) has a total of: 18 participants x 3 *techniques* { Default, 2-Fingers, BezelCopy } x 3 font sizes {15, 18, 21sp} x 4 text granularity {word, phrase, sentence, paragraph} x 2 trials per condition = 1296 trials.

Dependent variables were accuracy (ratio of successful trials to total trials), copy time (the interval between the start trial button and completion of copying the text to the clipboard), paste time (the interval between the end of copy time to finishing pasting the text to the target application), and total time (copy + paste time).

6.5 Results

We ran a 3-way repeated-measure ANOVA on **Technique**, **Granularity** and **Font size** for each dependent variable.

6.5.1 Accuracy

Accuracy was measured as the number of trials where users were able to copy the exact expected text to the target application out of the total number of trials. Overall, each technique achieved high selection accuracy. Traditional CP was the most accurate selection technique (99.3%), followed by the 2-Fingers (97.45%) and BezelCopy (96.29%). The ANOVA did not yield any significant differences for accuracy (p=.11).

6.5.2 Copy Time

We found a significant effect of **Technique** on copy time ($F_{2,34}$ = 19.14; p <.001). Pairwise t-Tests (with Bonferroni correction) showed significant differences between each technique (p<.001 for each test) with BezelCopy (5.84s) performing significantly faster than traditional method (7.61s) and 2-Fingers (10s). This suggests that BezelCopy is easier and faster to trigger since it does not rely on a long press that induces a delay, and despite the fact that the copy is divided in two steps: the bezel drag with the selection of one sentence/paragraph, then the adjustment of the selection on the new panel.

There were also a significant effect of **Granularity** on copy time ($F_{3,51}$=29.53; p<.001). Pairwise t-Tests indicated that copying a paragraph (9.92s) and phrase (9.77s) were significantly slower (p<.001) than copying a sentence (5.8s; p<.001) or word (5.79s; p<.001). We also found an interaction between **Technique and Granularity** ($F_{6,102}$= 4.04; p<.01). We can conclude that each of our technique has an optimal granularity where copying is faster. For traditional and 2-Fingers method, this granularity is **word**, while BezelCopy is optimal for **sentence**, as seen on Figure 3 (a).

The **Font size** factor also had a significant impact on the copy time ($F_{2,34}$=8.27; p<.01), with a significant difference (p<.05) between 15sp15sp (average copy time is 8.66s) and 18sp (7.1s). This suggests that 15sp15sp is harder to properly select, thus leading to longer copy times. Finally, we found an interaction **Granularity x Font size** ($F_{6,102}$= 16.15; p<.001). Both these factors have an impact on the size of the selection the user has to perform: the smaller the font is, the smaller the selection is. While a large font size makes the selection easier and quicker, a granularity such as sentence or paragraph leads to a large selection that takes more time. The interaction reflects this trade-off between precision and the quantity of information displayed.

6.5.3 Paste Time

As we explained previously, one of the main advantages of BezelCopy over the classical CP methods is that the selection adjustment panel also proposes a list of recently opened applications. It is then not surprising that the 3-way ANOVA yielded a significant difference of **Technique** on copy time ($F_{2,34}$=105.08; p<.001), with BezelCopy (3.01s) performing faster than 2-Fingers (6.80s; p<.001) and traditional (6.64s; p<.001). No any other significant differences for **Granularity** and **Font size,** or any interactions were found.

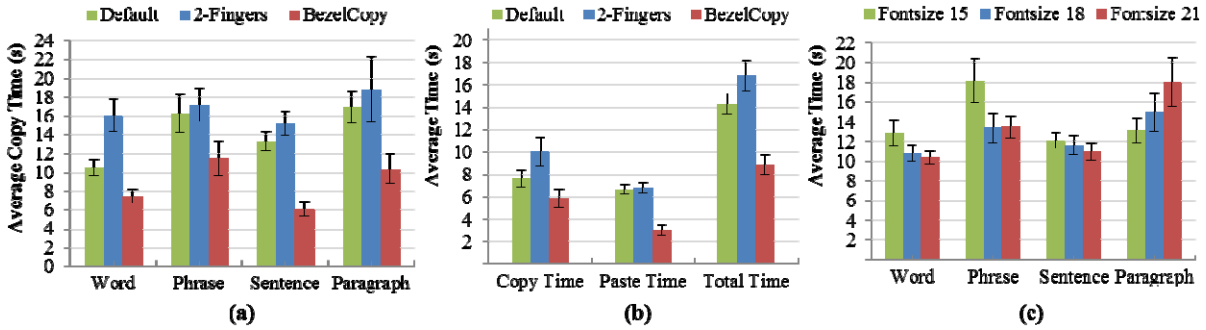

Figure 3: (a) Average copy time depending for each granularity and technique. (b) Average time to perform copy, paste and both operations depending on techniques. (c) Average time to perform a copy-paste operation depending on granularity and font sizes. Error bars are .95 confidence intervals.

6.5.4 Total Time

Figure 3(b) shows the copy, paste and total time to operate CP operations for each technique. Since total time is the addition of Copy and Paste time, it is not very surprising that the **Technique** had a significant impact on total time ($F_{2,34}$=65.64; p<.001). Pairwise t-Tests (with Bonferroni correction) showed that BezelCopy (average total time: 8.86s) is nearly two times faster than traditional (14.26s; p<.001) and 2-Fingers (16.81s). No differences were found between traditional and 2-Fingers (p>.05).

The **Granularity** factor also significantly impacted the total time ($F_{3,51}$=24.02; p<.001). Pairwise t-Tests yielded the same statistical differences than the one we found for copy time. An interaction between **Technique x Granularity** was also found ($F_{6,102}$=3.19; p<.001). These differences can exclusively be explained by the influence of these factors on copy time.

The 3-way ANOVA showed an effect of **Font size** ($F_{2,34}$=4.58; p=.02). While pairwise t-Tests did not find any difference between the three font sizes, a Tukey HSD found a slight yet significant difference between Font15 and Font18 (p<.05). The same **Granularity x Font size** was found for total time ($F_{6,102}$=13.48 ; p<.001). If we have a careful look at Figure 3-c, we can see that the **font size 18** is possibly an interesting size for selection, since the time to perform a CP operation with this value is either optimal (phrase granularity) or really close to optimal (for word, sentence & paragraph granularity).

6.6 Discussion

After the study, we interviewed the participants to understand how they felt with the traditional and 2-fingers methods.

6.6.1 Traditional method

Overall, users found that selection through this method was not suited for certain cases. The first case is the selection of long texts (i.e. when the text length exceeds what the screen can show), because they need to adjust selection and scroll multiples times.

Other concerns can be linked to the fat finger problem and the screen occlusion [19], e.g. the manipulation of selection handler that was deemed not really precise and sometimes unresponsive, making selection adjustments harder to perform; or selection with small font size. Users also suggested that switching between applications during a copy paste operation is time consuming.

Overall, the traditional Android method also has two drawbacks compared to other mobile OS: the paste command can only be activated when the user performs a long press on the touch screen, whereas Windows Phone users always have the paste button accessible on the keyboard; and the magnifier on iOS is considered more useful and precise than the selection handler.

6.6.2 2-fingers method

The 2-fingers method was also criticized by our participants. Selection difficulty with small font size was commonly observed and mentioned. Another encountered problem was that selecting the right text on the first try is really hard with this method, thus leading participants to adjust the handler most of the time.

This method suffers from screen occlusion problem even more than the traditional one, and participants pointed out that putting two fingers on the screen consumes too much space on it, which makes selection of short texts very complicated. There were also some hardware issues: in particular cases, the phone would not always detect the second finger, triggering the traditional selection method instead. Finally, the 2-fingers methods is not usable in every context, since it is in conflict with other two contacts gestures such as pinching from zooming. Thus, it is only usable in text editing contexts.

6.6.3 BezelCopy

Overall, participants are very positive towards BezelCopy. Participants stated that the auto enlargement on CP panel made text selection easier and more accurate. Also the auto-paste feature after selecting an application was highly appreciated, especially since no frustrating long press operation is needed. BezelCopy was also appreciated to copy sentences or paragraph, since the selection can be done with the first bezel drag gesture. In that case, the user only needs to select the application to paste the text into. Contrary to the two other methods, participants (10/18) reported to find it easier to initiate the copy operation even on small fonts, which suggests that the screen occlusion and fat finger are less annoying with BezelCopy.

However, they also pointed out a few points for improvement. Participants mentioned that BezelCopy is not necessary for copying isolated words as selecting it can be achieved with a simple tap with existing smartphones.

7. USER STUDY 2: BEZELSWIPE VS. BEZELCOPY

In addition to the currently supported CP techniques, we also wanted to compare the performances of BezelCopy with a previously proposed related technique BezelSwipe [14], since both techniques rely on the same bezel interaction mechanisms.

BezelSwipe also makes use of bezel drags for selection. The selection trigger is the partially same as in BezelCopy: user

performs a bezel drag from the left side of the screen to indicate the first word. To finalize selection, user then performs a bezel drag from the right side of the screen to indicate the last word, then the user can copy the text between the first word and the last word by pressing a "Copy" button on the top right of the screen. Paste operation is operated the same way as traditional Android methods. On one hand, BezelCopy proposes an efficient way to quickly paste between applications, while on the other hand BezelSwipe offers a finer consistency on selection: by finishing selection without changing.

7.1 Description of the study

We recruited 8 participants (4 females, aged from 22 to 30 average: 25.4) from our university for the study. In this study, we used the same Nexus S from the previous user study. All participants were asked to copy text from a source application and paste it into a different application as in the previous user study. We used the same four different corpuses of texts. In the experiment, we compared the performance of BezelSwipe and BezelCopy. In addition to the techniques, we considered font size and granularity. The experiment took 20 minutes to complete.

The study compares the time needed to operate a copy-paste operation depending on three factors: **Technique** {BezelCopy, BezelSwipe}, **Granularity** {word, phrase, sentence, paragraph} and **Font size** {15, 18, 21sp}. All these factors were counterbalanced between participants using Latin square.

To get familiar with the techniques, each participant had a 4 trials training phase with each technique and had to perform a complete copy-paste operation with all the text granularity conditions. Excluding these training phases, the study involved a total of 8 participants x 2 techniques x 3 font sizes x 4 text granularity x 2 trials per condition = 384 trials.

We measured four dependent variables: accuracy, time to perform copy, paste operation and the total time. To process the results, we ran three 2 x 4 x 3 repeated measures ANOVA.

7.2 Results

7.2.1 Accuracy

In conformity with the previous user study, participants did not do many errors. BezelSwipe (97.91%) was a bit more accurate than BezelCopy (96.87%). However, a paired t-Test on accuracy rate did not yield any significant differences (p=.28).

7.2.2 Copy Time

While BezelCopy and BezelSwipe have the same trigger, namely a bezel drag, selection with BezelSwipe is only down in one step. On the other hand, after a rough selection, users have to adjust on a magnified view, which could lead to a slower interaction. However, the 3-way ANOVA we ran did not yield any significant effect of **Technique** on copy time (p=.17). BezelCopy is actually slightly (no significance) faster (5.61s) than BezelSwipe (6.65s).

In accordance with results of the previous user study, we found a significant effect of **Granularity** ($F_{3,21}$=10.91, p<.001) on copy time. Pairwise t-Tests suggests than copying a word is faster (4.27s) than is slower than copying a paragraph (6.79s; p<.001); or a phrase (7.43s) or even a complete sentence (6.02s; p=.02). The same interaction **Technique x Granularity** ($F_{3,21}$=3.59; p<.01) was found, for the same reason as in the first user study: while BezelCopy's copy time is optimal for sentence, BezelSwipe

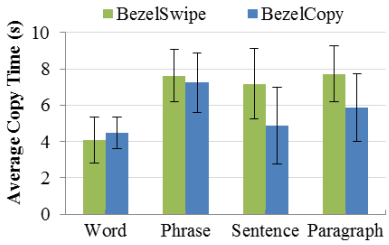

Figure 4: Time to perform CP operation for granularity and technique. Error bars are .95 confidence intervals.

is better for selecting words (Figure 4).

In the same way, **Font size** also had an impact on copy time ($F_{2,14}$=6.72; p<.01), with 15sp font size being harder thus longer to select (6.92s) that 18sp, (5.59s; p=.04). The same tradeoff between font size and length of the selected text was found, with an interaction **Granularity x Font size** ($F_{6,42}$=6.26; p<.001).

7.2.3 Paste time

As expected, the only factor having an effect on paste time was the **Technique** ($F_{1,7}$=71.99; p<.001), with BezelCopy (2.71s) performing better than BezelSwipe (6.17s) as shown on Figure 5.

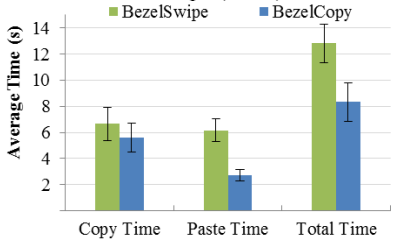

Figure 5: Time to perform copy, paste and both operations for each technique. Error bars are .95 confidence intervals.

7.2.4 Total time

The **Technique** had a significant impact on total time ($F_{1,7}$=30.7; p<.001; Figure 4). The performance of BezelCopy (8.32s) is comparable to its performance in the first user study. BezelSwipe, while being slower (12.81s) seems then to be a bit faster than the traditional and 2-Fingers technique.

In compliance with the previous user study, **Granularity** also had an impact on total performance ($F_{3,21}$=8.47, p<.001). Copy and pasting an isolated word is faster (8.84s) than copying a paragraph (11.22s; p=.02) and a phrase (11.77s; p<.01). Interestingly, there were no interaction Technique x Granularity for total time.

Also, total time depended on the **Font size** factor ($F_{2,14}$=7.61; p<.001). While pairwise t-Tests did not yield significant results, a Tukey HSD post-hoc test confirmed that 15sp is harder to copy-paste than 18sp (p<.01). The same interaction **Granularity x Font size** ($F_{6,42}$=6.08; p <.001) was found.

7.3 Discussion

Overall, the results of this user study confirmed the findings of the previous study. However, it is interesting to notice that *BezelCopy*, despite implying more steps to perform a correct selection, was slightly (yet not significantly) faster than *BezelSwipe*. The magnified panel for precise selection appears to be the quickest way to precisely copy text. An explanation for the slightly longest copy time for *BezelSwipe* is also the fact that many users reported troubles for finishing the selection: first, with

small font size, selection was deemed difficult, and also, in many cases, people would do a bezel drag from the left side of the screen, that would result in a new selection process instead of ending the current one. Since our two user studies have a very comparable experimental protocol, we can roughly compare the performance of the four tested techniques and see that bezel-based technique are more desirable for copy-paste operation than the one based on long press. The average accuracy of Bezel-based technique also shows that bezel gestures are quick and very precise for selection. With BezelCopy, we provide a convenient way to perform within- and cross-application CP operations.

8. IMPLICATIONS FOR DESIGN

The user studies we ran highlighted important features that designers should consider when designing mobile applications.

Optimal font size for mobile applications. Our results show a trade-off between font size and granularity, meaning that large font eases text selection, but it will increase the selection time for large granularity such as sentence or paragraph, since users then have to scroll over pages for these situations. Overall, the font size 18sp seems to be the best balance, since it allows for precise selection on low granularity and still performs well for high granularity levels. This particular font size is thus advised for designing a mobile application.

Optimization of CP techniques. In the user study, we observed varying performance between Technique and Granularity, which implies that each technique is better for a particular granularity. BezelCopy is better for Performance of word, phrase copy and paste can be different from sentence and paragraphs, so depends on how many do you expect the user will likely to copy, you can optimize for such type of CP. For example, the current selection of text in the CP panel of BezelCopy could be optimized by allowing the user to change the granularity of the selection (currently character) to make this step even easier.

Bezel gestures as a trigger for mode switching. We also recommend using bezel gestures over long press in order to trigger contextual actions. Long presses indeed induce a longer interaction time and also cause screen occlusion. Bezel gestures avoid all these drawbacks and are non-ambiguous. Bezel swipe is only one way to indicate copy mode, there are other ways that can be used, such as a button press, or touch the back of the device (anything that serve as a mode switching technique can be used).

9. CONCLUSION

We presented BezelCopy, a novel copy-paste technique for mobile devices. By relying on a bezel drag to trigger the copy operation, BezelCopy makes the selection of text easier and faster. The CP panel with magnified text allows for a quick and precise selection which lessens the fat finger problem. Moreover, the cross-application paste and auto-paste features make it possible to perform seamless cross-application paste operations. We compared BezelCopy with two traditional copy-paste techniques and BezelSwipe, and BezelCopy was proven up to two times faster than these techniques. Finally, we proposed guidelines to improve both copy-paste techniques on mobile devices, but more broadly the design of mobile applications.

10. ACKNOWLEDGEMENT

This work was supported by the Swedish Foundation for International Cooperation in Research and Higher Education (STINT, grant 2013-019) and National University of Singapore Academic Research Fund R-252-000-467-720.

11. REFERENCES

[1] Ashbrook, D., Lyons, K. and Starner, T. An investigation into round touchscreen wristwatch interaction. In *Proc. MobileHCI '08*. ACM, 311-314.

[2] Bier, E. A., Ishak, E. W. and Chi, E. Entity quick click: rapid text copying based on automatic entity extraction. In *CHI EA '06*. ACM, 562–567.

[3] Bragdon, A., Nelson, E., Li, Y. and Hinckley, K. Experimental analysis of touch-screen gesture designs in mobile environments. In *Proc. CHI '11*. ACM, 403-412.

[4] Chapuis, O. and Roussel, N. Copy-and-paste between overlapping Windows. In *Proc. CHI '07*, ACM, 201–210.

[5] Cockburn, A., Ahlström, D. and Gutwin, C. Understanding performance in touch selections: Tap, drag and radial pointing drag with finger, stylus and mouse. *IJHCS 70*, 3, 218–233, (2012).

[6] Faure, G., Chapuis, O. and Roussel, N. Power tools for copying and moving: useful stuff for your desktop. In *Proc. CHI '09*. ACM, 1675–1678.

[7] Fuccella, V., Isokoski, P. and Martin, B. Gestures and widgets: performance in text editing on multi-touch capable mobile devices. In *Proc. CHI '13*. ACM, 2785-2794.

[8] Ikematsu, K. and Siio, I. Memory stones: an intuitive copy-and-paste method between multi-touch computers. In *CHI EA '13*. ACM, 1287-1292.

[9] Ives, Z., Knoblock, C., Minton, S., et al. Interactive Data Integration through Smart Copy & Paste. In Proc. CIDR'09.

[10] Jain, M. and Balakrishnan, R. User learning and performance with bezel menus. In *Proc. CHI '12*. ACM, 2221-2230.

[11] Kim, S., Kim, J. and Lee, S. Bezel-flipper: design of a light-weight flipping interface for e-books. In *CHI EA* '13. ACM.

[12] Mistry, P., Nanayakkara, S. and Maes, P. Touch and copy, touch and paste. In *CHI EA* '11. ACM, 1095-1098.

[13] Ording, B., et al. Methods and Graphical User Interfaces for Editing on a Multifunction Device with a Touch Screen Display. *US Patent App. 12/565,750.* 2010.

[14] Roth, V. and Turner, T. Bezel swipe: conflict-free scrolling and multiple selection on mobile touch screen devices. In *Proc. CHI'09*. ACM, 1523-1526.

[15] Schmidt, D., Sas, C. and Gellersen, H. Personal clipboards for individual copy-and-paste on shared multi-user surfaces. In *Proc. CHI '13*. ACM, 3335-3344.

[16] Serrano, M., Lecolinet, E. and Guiard, Y. Bezel-Tap gestures: quick activation of commands from sleep mode on tablets. In *Proc. CHI '13*. ACM, 3027-3036.

[17] Shibuya, Y., Kawakatsu, H. and Murata, K. A web browsing method on handheld touch screen devices for preventing from tapping unintended links. In *Proc. HCI'13*. Springer.

[18] Stylos, J., Myers, B. A. and Faulring, A.. Citrine: providing intelligent copy-and-paste. In *UIST '04*. ACM, 185-188.

[19] Vogel, D. and Baudisch, P. Shift: a technique for operating pen-based interfaces using touch. In *Proc.CHI '07*. 657-666.

[20] Zhao, S., Chevalier, F., Ooi, W. T., Lee, C. Y. and Agarwal, A. AutoComPaste: auto-completing text as an alternative to copy-paste. In *Proc. AVI '12*. ACM, 365-372.