A Single-Handed Partial Zooming Technique for Touch-Screen Mobile Devices

Full papers

Jianwei Lai
University of Maryland,
Baltimore County
jianwei1@umbc.edu

Dongsong Zhang
University of Maryland,
Baltimore County
zhangd@umbc.edu

Sen Wang
University of Maryland,
Baltimore County
senwang1@umbc.edu

Abstract

Despite its ubiquitous use, the pinch zooming technique is not effective for one-handed interaction. We propose ContextZoom, a novel technique for single-handed zooming on touch-screen mobile devices. It allows users to specify any place on a device screen as the zooming center to ensure that the intended zooming target is always visible on the screen after zooming. ContextZoom supports zooming in/out a portion of a viewport, and provides a quick switch between the partial and whole viewports. We conducted an empirical evaluation of ContextZoom through a controlled lab experiment to compare ContextZoom and the Google maps' single-handed zooming technique. Results show that ContextZoom outperforms the latter in task completion time and the number of discrete actions taken. Participants also reported higher levels of perceived effectiveness and overall satisfaction with ContextZoom than with the Google maps' single-handed zooming technique, as well as a similar level of perceived ease of use.

Keywords

Zoom, single-handed, one-handed, mobile, touch-screen, context.

Introduction

Zooming is one of the most commonly performed operations on touch-screen mobile devices. An intuitive and smooth method of changing zooming levels is vital (Miyaki et al. 2009). Most of current touch-screen mobile devices allow users to zoom by finger pinch, which often requires both hands, with one hand holding a device and the other performing the pinch gesture. However, the pinch zooming technique does not work well for one-handed interaction, which is referred to as a user holding and interacting with a mobile device with the same hand simultaneously.

Prior research has suggested that most users prefer one-handed interaction, even when both hands are available (Karlson et al. 2007; Karlson et al. 2008), especially when they have situational impairments (Korhonen et al. 2007) or upper limb/hand disabilities (Lai et al. 2014). As opposed to a physical impairment, situational impairments or situationally-induced impairments (Sears et al. 2003) refer to impairments in which a user temporarily has difficulty in accessing mobile phones due to the context or situation she is in, which can be caused by environment context (e.g., bright sunlight), specific task context (e.g., driving a vehicle), or social context (disturbance caused to other people) (Korhonen et al. 2007). Situational impairments may occur when people us mobile phones on the go (Wobbrock 2006). For example, people carry bags while using mobile phones in a shopping mall. Current mobile devices with touch screens do not support one-handed interaction well (Park et al. 2010). In one-handed interaction, a hand has to secure a phone when its thumb performs other actions (Trudeau et al. 2012).

Due to the small screen size of mobile phones, users often lose context or get lost completely in the navigation space (Zhang et al. 2011). Zooming, especially zooming in, makes it difficult for a user to retain a sense of context and maintain a mental model of the navigation space (Qu et al. 2009; Robbins et al. 2004). Users often have a difficult time figuring out where they are on a webpage or map after zooming in.

To address the above problems, in this study, we design, develop, and evaluate a thumb moving direction based zooming technique called ContextZoom for one-handed interaction with touch-screen mobile devices. ContextZoom is aimed to provide sufficient context information to users when they perform one-handed zooming so as to avoid the problem of navigation loss. Here context includes user context (i.e., users' points of interest) and content context (i.e., the screen before zooming). ContextZoom is designed particularly for one-handed thumb use. It enables users to specify any location on a device screen as the center of zooming, so that the intended target (i.e., user context) will always stay in the viewport after zooming. The zooming level can be controlled by the traveling distance of a thumb on a device screen. ContextZoom supports partial viewport zooming. Users can see a portion of a screen in detail and go back to the previous screen before zooming quickly, so that users will not get lost during navigation.

The rest of the paper will be organized as follows. We will start with introducing the literature on existing zooming methods. Then, we will present the design of ContextZoom, followed by the description of the empirical evaluation methodology. Next, we will present results. Finally, the paper will be concluded with the discussion on major findings and limitations of ContextZoom.

Related Work

Multi-Finger Zooming Techniques

The two-finger pinch zooming technique (Jordà et al. 2010; Westerman 1999), the most popular zooming technique used for touch-screen mobile devices, often requires users to hold a device with one hand and perform pinch operations using the other by moving two fingers apart from or towards each other. Pinch-to-zoom is awkward and impractical for one-handed interaction, which is often needed when users have situational impairments, such as holding a bag in one hand (Ti et al. 2012), or have upper limb loss or disabilities. Double tap is also commonly used for single-handed zooming, with the first double tap to zoom in, and the second double tap to zoom out. Its limitation lies in that the zooming level is fixed. As a result, the double-tap method is not effective for tasks that require multiple zooming levels, such as browsing maps and photos. To address this problem, Google maps uses double-tap as a zooming gesture only, so that users can double tap multiple times to achieve multiple zooming levels. However, its zooming out motion requires users to tap with two fingers on the screen, which is challenging in one-handed interaction. In sum, existing multi-finger zooming techniques are not effective for single-handed interaction because they usually require two hands.

Single-Handed Zooming Techniques

With Google maps' single-handed zooming technique, a user first double taps on a device screen. Then instead of lifting the finger away from the screen, she drags her finger down/up on the screen to zoom in/out. Because this technique always zooms in/out with the current viewport center as the zooming center, when zooming in a map, a point of interest may go off the screen, causing potential confusion to users and extra effort to bring the interested location back to the viewport. Google maps also uses a pair of buttons ('+' for zooming in and '-' for zooming out) to change the zooming level, which also uses the viewport center as the zooming center. Thus, it suffers from the same problem as mentioned above.

GraspZoom (Miyaki et al. 2009) enables users to magnify content with one hand with the help from an external pressure sensor attached to the back of a mobile phone, which limits its practicality and adoption (Ti et al. 2012). A rubbing gesture, a small repetitive diagonal motion with a finger, is used by (Olwal et al. 2008) for zooming on touch-screen devices. This technique takes into account the orientation of the rubbing gesture. A right-handed user zooms in by rubbing back and forth along the lower-left-to-upper-right diagonal, and zooms out by rubbing along the lower-right-to-upper-left diagonal. The motion directions for left-handed users are opposite. CycloStar (Malacria et al. 2010) is another gesture-based technique, in which a user performs a circular gesture to zoom in (clockwise) or out (counter-clockwise). Fat Thumb (Boring et al. 2012) uses contact area size of a thumb tip to activate the zooming mode. A user moves a thumb with a small contact size to pan the content. By increasing the contact size, the user activates the zooming mode, with moving the thumb around its joint to the right for zooming in, or to the left for zooming out. Users may accidently switch to the zooming mode when panning. Using Fat Thumb may also require more cognitive efforts to control the contact size when moving a thumb on the screen. All those existing methods are challenging for single-handed zooming because some of the places on a mobile screen, such as corners, are difficult to reach with a thumb in single-handed interaction. Even with extra panning, some places are still difficult to reach, such as the address bar of an Internet browser.

Different from gestures performed on device screens, tilting gesture-based zooming methods tilt mobile devices in different ways for zooming, such as Tilt-to-zoom (Hinckley et al. 2011) and TiltZoom (Ti et al. 2012). Tilting a device towards (or away from) a user can zoom in (or out). Because a tilting gesture changes the angle of a device, it may prohibit the user from seeing the screen clearly during the motion.

Focus & Context Techniques

The zooming interaction supports both focused and contextual views with a temporal separation between them (Cockburn et al. 2009). Users either magnify (zoom in for a focus) or de-magnify (zoom out for context) an interface, but they can't see both views simultaneously (Cockburn et al. 2009). ZoneZoom (Robbins et al. 2004) divides a viewport into nine segments, each mapped to a key on a number keypad of mobile phones. User can zoom in a segment by pressing the corresponding key, and zoom out to the previous overall view by pressing the same key again or a dedicated "zoom-out" key on the keypad. Although ZoneZoom enable users to switch between focus and context views, it is not convenient for mobile devices without a physical keyboard, because bringing up the virtual keyboard requires an extra step in the first place. In addition, each key press can only make the view zoom in/out by a fixed level, which makes it impossible to reach any arbitrary zooming levels.

Focus-plus-context visualization techniques, e.g., Fisheye (Bederson et al. 2004), Flip zooming (Björk 2000) and AppLens (Karlson et al. 2005), display a focused area with a larger zooming level embedded in a surrounding context area with a smaller zooming level to fit into the available screen space, thus allowing users to examine information without losing context (Zhang et al. 2011). However, because an overview or surrounding area (i.e., context) is typically displayed in a much smaller zooming level or font size in comparison to the focused area, such context could be illegible on the screen of mobile devices and becomes useless, while wasting valuable display space (Zhang et al. 2011). In addition, although context information can help prevent users from getting lost during zooming, existing focus-plus-context visualization techniques do not work for zooming on mobile phones because they usually magnify the focused areas with fixed zooming levels.

In sum, although there exist a number of different single-handed zooming techniques, they have various limitations that hinder their effectiveness: 1) by using the screen center as the zooming center, a target can go off screen after zooming in; 2) due to zooming with fixed zooming levels, users cannot reach any zooming levels arbitrarily; and 3) there is a lack of context information provided to users during zooming.

Design of ContextZoom

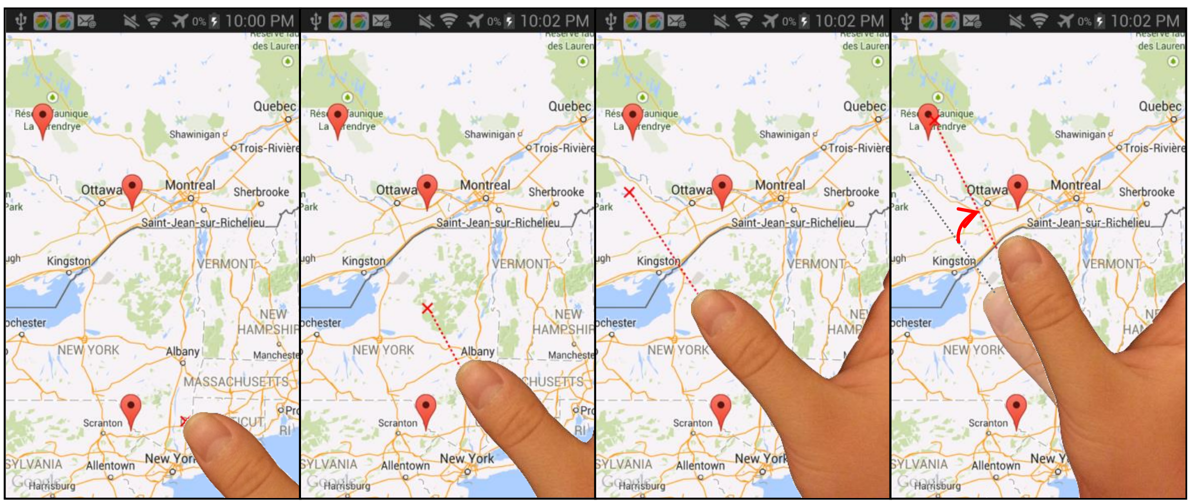
In this research, we have designed and evaluated ContextZoom, a new single-handed zooming technique for touch-screen mobile handheld devices, which is aimed to address the limitations of existing methods described above. The design of ContextZoom takes several issues into consideration. Specifically, ContextZoom is designed to

- support single-handed zooming;
- allow a user to specify any point on a mobile device screen of her interest as the zooming center and ensure that the interested target will still be visible on the device screen after zooming in;
- enable selection of any distant target as the zooming center; and
- present context of a navigation space to help users maintain a mental model of the navigation space to prevent them from getting lost.

Google maps uses a long finger press on a touch screen to select a point on the screen as the point of interest and insert a pin at that location. To be consistent with Google maps, we also use a long press in

ContextZoom for users to specify a point as the zooming center, and all targets are presented as pins. With ContextZoom, a long press on an existing pin will select that pinned location as the zooming center.

One of the challenges in single-handed interaction is target selection (Lai et al. 2014). The morphological constraints of a thumb limit its movements. As a result, some places on a device screen may be difficult to reach (Karlson et al. 2005). In addition, a thumb touching on a device screen will occlude the content underneath, thus making target selection error-prone (Scheibel et al. 2013). For difficult-to-reach targets, although the user can perform panning to move distant targets towards her thumb, other targets may go off the screen. To address this target acquisition problem, we adapt ExtendedThumb (Lai et al. 2014) to enable a user to reach and select difficult-to-reach objects with one hand without panning. ExtendedThumb works in the following way: after a long press on the screen, without lifting the thumb away from the screen, the user moves the thumb on the screen toward a target (Figure 1). A red cross, which is controlled by the real thumb, moves in the exactly same direction (indicated by the dotted line) as the real thumb but with a longer moving distance. A 2:1 default moving distance ratio between the red cross and the real thumb is used in this study. For nearby pins/targets, a user can perform a long press on them with her thumb directly to select them.



(a) (b) (c) (d)

Figure 1. Zooming Center Selection: (a) a red cross appears initially at the point underneath the thumb after a long press. If the thumb leaves the screen, the current pressed location will be selected as the zooming center. Otherwise, (b) when the thumb moves towards a target, the red cross goes in the same direction but with doubled traveling distance; (c) the red cross misses the target; and (d) the user adjusts the position of the red cross by turning the thumb and moving it towards the pin. When the red cross is on the pin, the user can select it as the zooming center by lifting the thumb up from the screen.

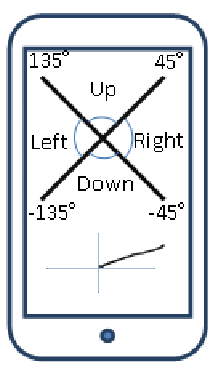
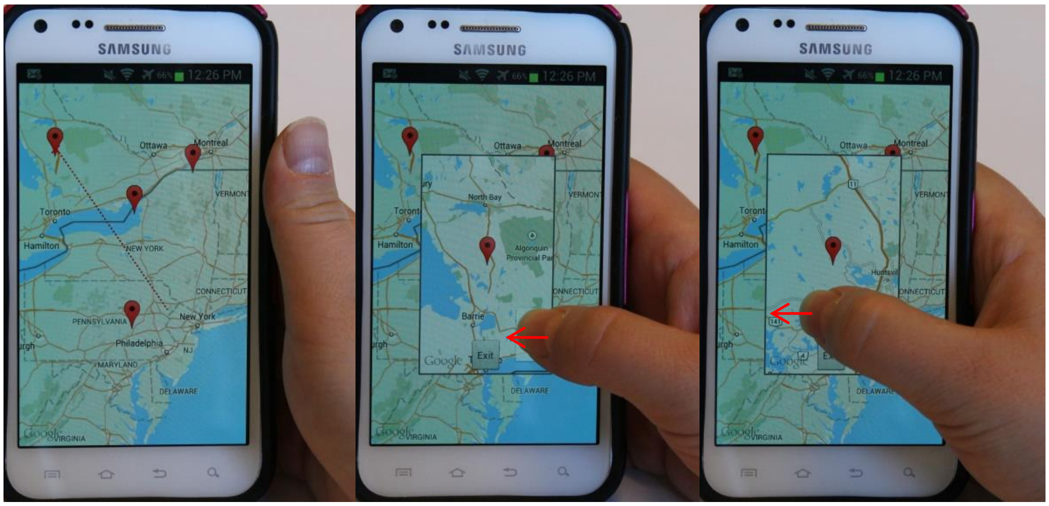


Figure 2: Ranges of Four Moving Directions

After selecting the zooming center, a user can move her thumb from anywhere on the screen to perform zooming or do a long press on another location to re-select a different zooming center. As long as the red cross is on the screen, the panning function is disabled. After zooming is performed, the red cross will disappear and the panning function will be automatically resumed. In ContextZoom, there can be four directions of users' thumb movement, i.e., up, down, left, and right (Figure 2).



(a) (b) (c)

Figure 3. Partial Zooming

A user can move her thumb from anywhere on the screen to the left to zoom in a partial viewport (Figure 3 (b)), with the selected zooming center at its center and a default size being 1/3 of the overall viewport. The user can reset the size of the partial viewport. The longer the user drags her finger to the left, the more a map will zoom in (Figure 3(c)). When the thumb leaves the screen, the partial zooming mode will be aborted. The user can interact with a map in a partial viewport in the same way as interacting with a regular map. For example, she can perform the Google maps' single-handed zooming technique in the partial viewport to adjust the zooming level. The user can exit the partial viewport and go back to the original display of the map (as shown in Figure 3(a)) by clicking the "Exit" button at the bottom of the partial viewport. Similarly, users can drag a thumb to the right on a device screen to zoom out a portion of a map after selecting a zooming center.

Implementation

A prototype of ContextZoom has been implemented in Java using the Android SDK and Google maps' APIs in Eclipse for user evaluation. The prototype system was installed on a Samsung Galaxy S2 phone featuring a 1.2GHz dual-core processor, a 4.52" Super AMOLED Plus (480*800 pixels) display, 1GB RAM, and the Android 4.1.2 operating system. A system log in this phone automatically records the time of interactions when a user uses ContextZoom.

Evaluation

A controlled laboratory experiment has been conducted to evaluate ContextZoom, with the Google maps' single-handed zooming method (referred as GMS hereafter) as a baseline method. We chose GMS because Google maps is commonly used by users. Although double-tap is used by Google maps as a zoom-in gesture, the corresponding zoom-out motion requires a two-finger-tap, which is challenging in one-handed interaction. Thus, we did not include the double-tap zooming technique in the evaluation.

Within the partial viewport of ContextZoom, users can interact with a map like regularly, such as panning, zooming, and clicking. To minimize the confounding effect, we only allowed users to use GMS within the partial viewport. In other words, ContextZoom was used in the study with the GMS technique enabled in the partial viewport (referred to as ContextZoom+GMS hereafter). Panning was allowed within both the whole and partial viewports.

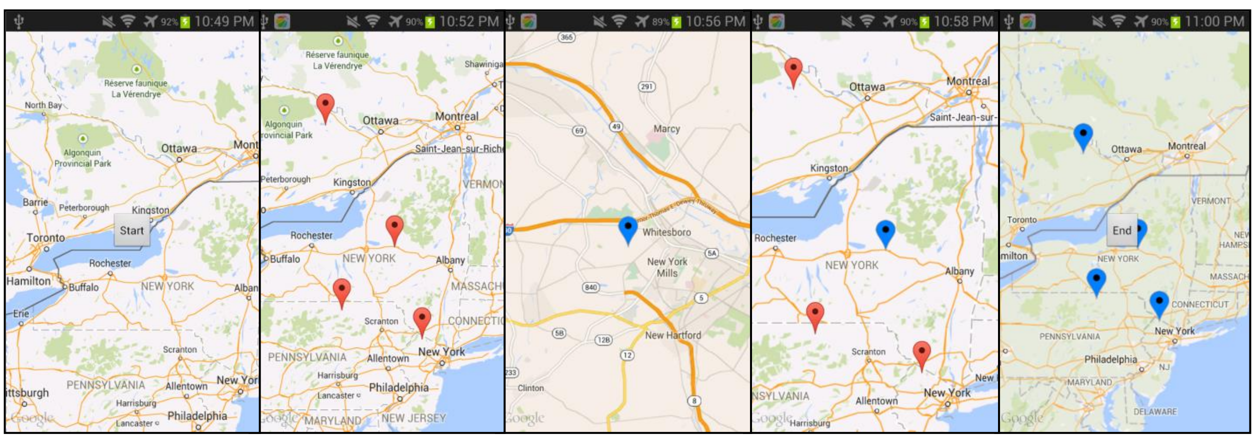
Participants

23 participants (13 male, 10 female) at an east-coast university in the United States were recruited to participate in the study. They were undergraduate and graduate students with a major in information systems. Among them, 15 were between 18 and 25 years old; 7 were between 26 and 30 years old; and 1 was over 30 years old. They were all right-handed and had prior experience with touch-screen mobile phones. Participants received extra course credits for participating in the experiment.

Procedure

In this study, to simulate a situational impairment condition, the participants carried out six zooming tasks (Figure 4) while walking on a treadmill. They were required to hold a Samsung Galaxy S2 phone and interact with it with their dominant hand only, while holding an empty water bottle in the other hand. By following a previous study (Bergstrom-Lehtovirta et al. 2011), the moving speed of the treadmill was set by participants according to their normal walking speed, while interacting with a touch-screen device. The mean of participants' walking speed was 1.9 km/h (SD=0.8 km/h).

After signing a consent form, participants went through a training session prior to the experiment. After participants were comfortable with the experimental zooming methods and tasks, the experiment would start.



(a) (b) (c) (d) (e)

Figure 4. Zooming Tasks

Once a participant clicked the start button on the system interface (Figure 4(a)), two, four, or eight targets would appear on the screen as red pins (Figure 4(b)). The system recorded this time as the starting time. A participant was instructed to zoom in a map until the pins turned into blue. When the zooming level of the map was equal to or higher than a certain level (referred to as the aimed zooming level), a red pin in the viewport would turn into blue (Figure 4(c)). Pins would not turn into blue if they were off the viewport. The locations of the pins were randomly picked on the screen, but targets were at least 100 pixels away from each other when they initially appeared on the screen to guarantee that only one target appeared within the viewport after a participant reached the aimed zooming level. Participants needed to zoom out (or go back to the overall viewport by clicking the "Exit" button in the partial viewport with the ContextZoom+GMS method) to find another target (Figure 4(d)), and then zoomed in to reach the aimed zooming level as quickly as possible. After all targets turned into blue, participants needed to zoom out or go back to the overall viewport. When a map's zooming level is equal to or lower than the initial zooming level, a "Next" button would appear on the screen. The system recorded the time when the "Next" button appeared as the ending time of the current zooming task. By clicking the "Next" button (Figure 4(e)), participants could start the next zooming task. Participants were instructed to complete the zooming tasks as fast as possible. The initial zooming level and aimed zooming level were the same for all conditions.

When users search in maps, such as searching for nearby restaurants, they usually get different numbers of results. Thus, we used three different target numbers, namely two, four, and eight, to represent low, medium, and high target density. The sequences of zooming techniques and the target density levels were balanced among individual participants in order to minimize learning effects. Participants finished three tasks (each with a different target density) using GMS and ContextZoom+GMS. Thus, each participant finished a total of six tasks.

Independent and Dependent Measures

The independent variables are the zooming method and target density. There were two zooming methods, i.e., GMS and ContextZoom+GMS. Target density included three levels: two, four and eight.

Dependent variables include participants' performance and perception. Participants' performance of zooming tasks was assessed through task completion time and number of discrete actions:

- Task completion time was measured as the duration between the time when targets appeared on the device screen and the time when a participant went back to the initial zooming level after all targets turned into blue.
- Number of discrete actions: the experimental system recorded each discrete action performed on the device screen by individual participants during each task, including tap and moving actions. The combined number of tap and moving actions was counted as the discrete action number.

Perceived ease of use, perceived effectiveness, and overall satisfaction of participants with regard to zooming methods were assessed through seven 7-point Likert scale questions (Table 1). Those questions were adapted from the IBM Post-Study System Usability Questionnaire (Lewis 1995) and were grouped into three factors. The questionnaire was developed based on the application of psychometric methods to measure user satisfaction and subjective assessment with system usability (Lewis 1995). It is believed that mental workload is an appropriate measure of usability when a task involves continuous demand of a user's attention for monitoring and control (Lewis 1995).

Factors	**Items** (1 representing "Totally Disagree" and 7 representing "Totally Agree")
Perceived ease of use	Overall, I am satisfied with how easy it is to use this method. It was simple to use this method. It was easy to learn to use this method.
Perceived effectiveness	I could effectively complete the tasks using this method. I was able to complete the tasks quickly using this method. I was able to efficiently complete the tasks using this method.
Overall satisfaction	Overall, I am satisfied with this method for target selection.

Table 1. Questions Measuring User Perceptions

Results

Task Completion Time

The means of task completion time are shown in Figure 5. Task completion time were analyzed using 2*3 repeated measures analysis of variance with within-subjects factors of zooming method (ContextZoom+GMS and GMS) and target density (two, four and eight). Mauchly's test of sphericity showed no violations. The main effect of zooming method ($F(1, 22) = 96.15$, $p < 0.001$) is significant, but the main effect of target density (F (2, 44) = 2.34, p > 0.05) and the interaction effect between them was insignificant (F (2, 44) = 3.21, p > 0.05). Bonferroni Post-hoc test results show that the task completion time of ContextZoom+GMS is significantly less than that of GMS (mean difference = -4.68, p < 0.001). Participants using ContextZoom+GMS took significantly and consistently less task time than the counterparts when using GMS only at all three levels of target density. Paired samples t-test results are t = 8.90, 5.70, and 5.54, respectively (p < 0.001).

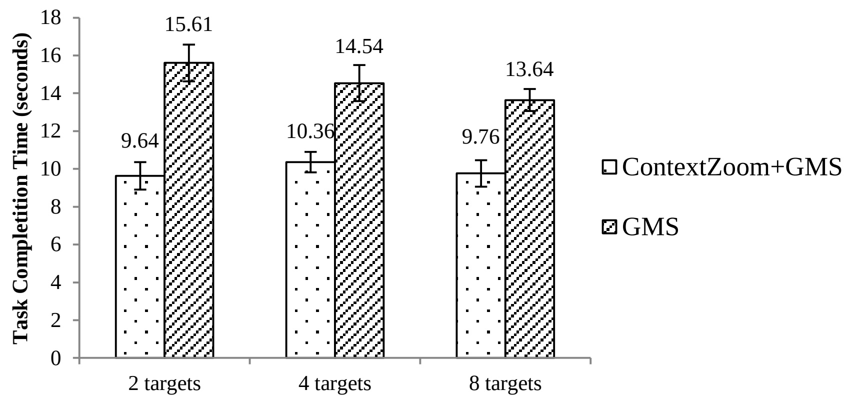


Figure 5. Means of Task Completion Time (error bars representing standard errors)

Number of Discrete Actions

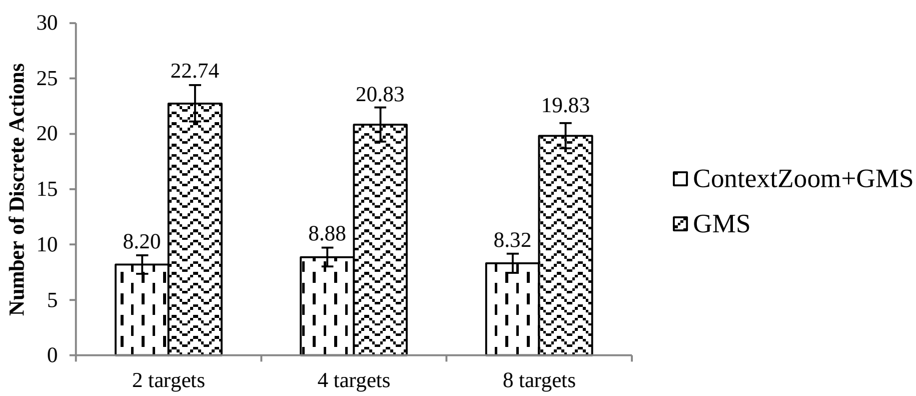


Figure 6. Discrete Action Numbers (error bars representing standard errors)

The means of discrete action numbers are shown in Figure 6. The number of discrete actions is analyzed using 2*3 repeated measures analysis of variance with within-subjects factors of zooming method (ContextZoom+GMS and GMS) and target density (two, four and eight). Mauchly's test of sphericity showed no violations. The main effects of zooming method (F (1, 22) = 151.95, p < 0.001) and the interaction effect between zooming method and target density (F (2, 44) = 3.92, p < 0.05) are significant, but the main effect of target density is not (F (2, 44) = 2.81, p > 0.05). Bonferroni Post-hoc test results present that the number of discrete actions for ContextZoom+GMS is significantly smaller than that of GMS (mean difference = -12.67, p < 0.001). Participants performed significantly fewer discrete actions with ContextZoom+GMS than with GMS only at all three levels of target density. Paired samples t-test results are t=11.00, 10.01, and 9.77, respectively (p < 0.001).

User Perceptions

The Cronbach's Alphas for perceived ease of use and perceived effectiveness constructs were 0.87 and 0.96, respectively, all above the recommended minimum level (Peterson 1994).

Factor	Method	Mean	SD	t
perceived ease of use	ContextZoom+GMS	5.67	1.19	1.49
	GMS	5.09	1.61	
perceived effectiveness	ContextZoom+GMS	5.81	1.45	2.33*
	GMS	4.70	1.64	
overall satisfaction	ContextZoom+GMS	5.96	1.61	5.99***
	GMS	2.91	1.61	

*p < 0.05; *** p < 0.001

Table 2. Means of Perceived Ease of Use, Perceived Effectiveness, and Overall Satisfaction

Results in Table 2 demonstrate that participants reported higher levels of perceived effectiveness ($p < 0.05$) and overall satisfaction ($p < 0.001$) with ContextZoom+GMS than with GMS only, while achieving a similar level of perceived ease of use.

Discussion

ContextZoom makes several contributions to the research on single-handed zooming on mobile handheld devices. First, ContextZoom improves users' one-handed zooming performance. Second, when a point of interest is selected as the zooming center, it will be guaranteed to stay in the viewport during zooming, so that the user will not miss it. Third, by incorporating the adapted ExtendedThumb technique, ContextZoom enables users to select a target point as the zooming center by moving her thumb on the screen toward the target after a long press. A red cross reaches out along the moving direction of the real thumb with a longer distance. When the user's thumb leaves the screen, the point underneath the red cross will be chosen as the zooming center. By adapting ExtendedThumb, users can reach remote objects on a device screen without panning. Meanwhile, a selected point is not under the thumb but at the point of the red cross. As a result, the visual occlusion problem (i.e., the user is uncertain about which point underneath the thumb is the activation point (Boring et al. 2012; Roudaut et al. 2008; Scheibel et al. 2013)) can be avoided. Fourth, ContextZoom enables the user to switch between the whole viewport and the partial viewport, so that the user can quickly go back to the original screen after zooming in the partial viewport by clicking the "Exit" button. It is especially helpful when there are multiple targets. The partial view provides users with a portion of content in detail without losing the overall context e.g., the locations of other targets.

ContextZoom outperformed GMS in both task completion time and number of discrete actions. It may be because the baseline technique always uses the viewport center as the zooming center, in which some targets may go off the screen after zooming. Therefore, users have to find and bring those targets back to the viewport, and they may get lost. With ContextZoom, the user can make a target as the zooming center to make sure it stays in the viewport after zooming. Even if the user accidently moves a target originally at the center out of the partial viewport, she can easily go back to the initial screen in the whole viewport to start over quickly. Furthermore, with GMS, participants had to zoom out to find other targets after finishing one. In contrast, with ContextZoom, participants could go back to the starting screen in the whole viewport by clicking the "Exit" button of the partial viewport. This design could save time and reduce operations.

For the number of discrete actions, the interaction effect between zooming method and target density is significant. As the target density increased, the number of discrete actions of GMS decreased. It may be because for a task involving more targets, participants needed to zoom in/out for more times. As a result, they had more opportunities to learn the locations of targets.

Participants also achieved significantly higher levels of perceived effectiveness and overall satisfaction when they used ContextZoom than when they used GMS, while achieving a similar level of perceived ease of use. It may be because the double-tap-hold gesture used in GMS is relatively unfamiliar. Users tend to lift their thumb away from the screen after the second tap, rather than to hold it on the screen. In fact, this double-tap-and-lifting gesture is not unusual. It has also been used in Google maps. Participant' previous experience with it may hinder them from learning the Google maps' single-handed zooming technique, which is indicated by participants' comments after the study that "double tap and hold is confusing", "it was awkward at first", and "it took time to get used to it".

The method to activate an interaction technique should not influence users' performance (Yu et al. 2013). Currently, a long press gesture is used to initiate zooming center selection and to activate the zooming mode of ContextZoom. Although the long press gesture can be incorporated in Google maps because the zooming center can be considered as a target and Google maps enables users to choose a target with a long press to drop a pin, this initiation method may conflict with other applications. We plan to explore other gestures, such as rubbing (Olwal et al. 2008) and bezel gestures (Bragdon et al. 2011), in a future study to minimize potential conflicts. Another limitation of ContextZoom is that when a user moves her thumb to zoom, she has to stop after reaching the furthest point on the screen that her thumb can reach. If she wants to change the zooming level further, she has to perform zooming in the partial viewport. The GMS technique shares a similar problem: after reaching the furthest location that her thumb can reach, the user has to repeat the double-tap and hold gesture to do further zooming, which could be tedious and time consuming. In the future, it is worth refining ContextZoom to overcome the limitations of thumb length and screen size.

Acknowledgements

This material is based upon work supported by the National Science Foundation (Award #: IIS-1250395). Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

Bederson, B. B., Clamage, A., Czerwinski, M. P., and Robertson, G. G. 2004. "DateLens: A fisheye calendar interface for PDAs," *ACM Transactions on Computer-Human Interaction* (11:1), pp. 90-119.

Bergstrom-Lehtovirta, J., Oulasvirta, A., and Brewster, S. Year. 2011. "The effects of walking speed on target acquisition on a touchscreen interface," in *Proceedings of the 13th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM, pp. 143-146.

Björk, S. Year. 2000. "Hierarchical flip zooming: enabling parallel exploration of hierarchical visualizations," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ACM, pp. 232-237.

Boring, S., Ledo, D., Chen, X. A., Marquardt, N., Tang, A., and Greenberg, S. 2012. "The fat thumb: using the thumb's contact size for single-handed mobile interaction," in *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM, pp. 39-48.

Bragdon, A., Nelson, E., Li, Y., and Hinckley, K. 2011. "Experimental analysis of touch-screen gesture designs in mobile environments," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 403-412.

Cockburn, A., Karlson, A., and Bederson, B. B. 2009. "A review of overview+detail, zooming, and focus+context interfaces," *ACM Computing Surveys* (41:1), pp. 1-31.

Hinckley, K., and Song, H. 2011. "Sensor synaesthesia: touch in motion, and motion in touch," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 801-810.

Jordà, S., Julià, C. F., and Gallardo, D. 2010. "Interactive surfaces and tangibles," *XRDS: Crossroads, The ACM Magazine for Students* (16:4), pp. 21-28.

Karlson, A., and Bederson, B. 2007. "ThumbSpace: generalized one-handed input for touchscreen-based mobile devices," in *Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction*, pp. 324-338.

Karlson, A. K., Bederson, B. B., and Contreras-Vidal, J. 2008. "Understanding one handed use of mobile devices," *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, pp. 86-101.

Karlson, A. K., Bederson, B. B., and SanGiovanni, J. 2005. "AppLens and launchTile: two designs for one-handed thumb use on small devices," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 201-210.

Korhonen, H., Holm, J., and Heikkinen, M. 2007. "Utilizing sound effects in mobile user interface design," in *Proceedings of the 11th IFIP TC 13 international conference on Human-computer interaction*, pp. 283-296.

Lai, J., and Zhang, D. 2014. "ExtendedThumb: a motion-based virtual thumb for improving one-handed target acquisition on touch-screen mobile devices," in *Proceedings of the Extended Abstracts of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 1825-1830.

Lewis, J. R. 1995. "IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use," *International Journal of Human-Computer Interaction* (7:1), pp. 57-78.

Malacria, S., Lecolinet, E., and Guiard, Y. 2010. "Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: the cyclostar approach," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 2615-2624.

Miyaki, T., and Rekimoto, J. 2009. "GraspZoom: zooming and scrolling control model for single-handed mobile interaction," in *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Serv*ices, ACM, Article 11.

Olwal, A., Feiner, S., and Heyman, S. 2008. "Rubbing and tapping for precise and rapid selection on touch-screen displays," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp. 295-304.

Park, Y. S., and Han, S. H. 2010. "One-handed thumb interaction of mobile devices from the input accuracy perspective," *International Journal of Industrial Ergonomics* (40:6), pp. 746-756.

Peterson, R. A. 1994. "A meta-analysis of Cronbach's coefficient alpha," *Journal of consumer research*, pp. 381-391.

Qu, H., Wang, H., Cui, W., Wu, Y., and Chan, M.-Y. 2009. "Focus+ context route zooming and information overlay in 3D urban environments," *Visualization and Computer Graphics, IEEE Transactions on* (15:6), pp. 1547-1554.

Robbins, D. C., Cutrell, E., Sarin, R., and Horvitz, E. 2004. "ZoneZoom: map navigation for smartphones with recursive view segmentation," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ACM, pp. 231-234.

Roudaut, A., Huot, S., and Lecolinet, E. 2008. "TapTap and MagStick: improving one-handed target acquisition on small touch-screens," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ACM, pp. 146-153.

Scheibel, J.-B., Pierson, C., Martin, B., Godard, N., Fuccella, V., and Isokoski, P. 2013. "Virtual Stick in Caret Positioning on Touch Screens," in *Proceedings of the 25th IEME Conference Francophone on l'Interaction Homme-Machine,* ACM, pp. 107-114.

Sears, A., Lin, M., Jacko, J., and Xiao, Y. 2003. "When computers fade: Pervasive computing and situationally induced impairments and disabilities," *HCI International*, pp. 1298-1302.

Ti, J., and Tjondronegoro, D. 2012. "TiltZoom: tilt-based zooming control for easy one-handed mobile interactions," in *Proceedings of the Internet of Things Workshop, OZCHI 2012: Integration, Interaction, Innovation, Immersion, Inclusion*, ACM: Melbourne, Victoria, Australia. ACM.

Trudeau, M. B., Udtamadilok, T., Karlson, A. K., and Dennerlein, J. T. 2012. "Thumb motor performance varies by movement orientation, direction, and device size during single-handed mobile phone use," *Human Factors: the Journal of the Human Factors and Ergonomics Society* (54:1), pp. 52-59.

Westerman, W. 1999. *Hand tracking, finger identification, and chordic manipulation on a multi-touch surface*, University of Delaware. PhD Thesis.

Wobbrock, J. O. 2006. "A robust design for accessible text entry," *ACM SIGACCESS Accessibility and Computing* (84), pp. 48-51.

Yu, N.-H., Huang, D.-Y., Hsu, J.-J., and Hung, Y.-P. 2013. "Rapid selection of hard-to-access targets by thumb on mobile touch-screens," in *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM, pp. 400-403.

Zhang, D., and Lai, J. 2011. "Can convenience and effectiveness converge in mobile web? a critique of the state-of-the-art adaptation techniques for web navigation on mobile handheld devices," *International Journal of Human-Computer Interaction* (27:12), pp. 1133-1160.