Two New Gestures to Zoom: Enhancing Online Maps Services

Alessio Bellino
DISCo, University of Milano-Bicocca
Viale Sarca 336/14 Milan, Italy
bellino@disco.unimib.it

ABSTRACT
Online services such as Google Maps or Open Street Maps allow the exploration of maps on smartphones and tablets. The gestures used are the pinch to adjust the zoom level and the drag to move the map. In this paper, two new gestures to adjust the zoom level of maps are presented. Both gestures – with slight differences – allow the identification of a target area to zoom, which is enlarged automatically up to cover the whole map container. The proposed gestures are added to the traditional ones (drag, pinch and flick) without any overlap. Therefore, users do not need to change their regular practices. They have just two more options to control the zoom level. One of the most relevant and appreciated advantages has to do with the gesture for smartphones (Tap&Tap): this allows users to control the zoom level with just one hand. The traditional pinch gesture, instead, needs two hands. According to the test results on new gestures in comparison with the traditional pinch, 30% of time is saved on tablets (Two-Finger-Tap gesture) whereas 14% on smartphones (Tap&Tap gesture).

Categories and Subject Descriptors
D.2.2 [**Design Tools and Techniques**]: User interfaces

General Terms
Human Factors, Experimentation.

Keywords
Zoom, Map, Navigation, Pinch, Multi-touch, Gesture, Smartphone, Tablet

1. INTRODUCTION
Currently, one of the most common activities performed by users in their mobile devices is the exploration of large 2D spaces such as maps, pictures and websites. In these devices, the use of zoom and drag is frequent because they usually have small screens. One of the most used multi-touch paradigms devoted to the 2D navigation is the Pinch-Drag-Flick because of its simplicity and effectiveness: pinch to zoom and drag-flick to pan. In spite of these advantages, the paradigm presents some drawbacks: fingers hinder the screen visualization [7] and the pinch is difficult to perform with just one hand when it comes to smartphones [6]. These disadvantages have fostered the development of several alternatives to navigate which, for instance, use different gestures [1] or extend the interaction to the side [5] or the back of the device [7]. In this paper two new gestures are presented, which are added to the traditional paradigm Pinch-Drag-Flick without replacing it. Furthermore, they do not require specific hardware. One gesture is designed for tablets whereas the other for smartphones. Particularly, the latter allows the usage of smartphones with just one hand, unlike the traditional Pinch-to-Zoom in which both hands are required: one to hold the smartphone and the other to act on the screen.

2. LITERATURE REVIEW
In this paper, an integration to the Pinch-Drag-Flick paradigm is designed, so that some details related to the latter are provided when the new gestures are explained (section 3.3). This section, instead, is focused on some alternatives to the traditional Pinch-Drag-Flick paradigm: the Slider, the CicloStar [4] and the Fat Thumb [1]. The first one, coming from desktop computers, is not common in touch devices. However, it allows the zoom adjustment by moving the slider upwards or downwards. The second [4] works by moving the finger in circles on the device screen: clockwise to zoom in and anticlockwise to zoom out. The third [1] lets users adjust the zoom by means of the contact area of their thumbs with the screen: with slight pressure, the contact area is small and the pan is allowed by moving the finger. On the contrary, with high pressure, the contact area is larger and the zoom adjustment is allowed. In this case, the zoom speed depends on the size of the contact area of the thumb. All these paradigms allow the zoom usage with just one hand. In [1] a comparison among Tilt-to-zoom [3], Slider, CycloStar [4] and Fat Thumb [1] was made and the latter is the faster. Some commercial software for desktop computers implement another way of zoom quite interesting for this research: identifying an area by means of selection, it will be enlarged up to cover the dimension of the window container. This zoom is called in different ways: "Marquee Zoom" in Adobe Acrobat X[1] and Zoom "Window" in Autodesk Autocad[2]. Moreover, a patent that uses a similar zoom principle was registered [2]. The new gestures allow the zoom adjustment on smartphones and tablets using an approach similar to the aforementioned zoom for desktop. In this literature review, spatial input-based interaction techniques were not taken into account because they need additional hardware (e.g. [6]). In addition, they require technologies far from Web standards.

WWW'15 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742823

[1] http://help.adobe.com/en_US/acrobat/X/standard/using/WS58a04a822e3e50102bd615109794195ff-7fd5.w.html

[2] http://knowledge.autodesk.com/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/AutoCAD-Core/files/GUID-66E7DB72-B2A7-4166-9970-9E19CC06F739-htm.html

3. NEW GESTURES

Two new gestures were designed: Tap&Tap for smartphones and Two-Finger-Tap for tablets. Both gestures are added to the traditional Pinch-Drag-Flick paradigm without replacing it, enhancing the possibilities of choice for users.

A demo video can be watched on http://youtu.be/3qaYs_hZSlQ

3.1 Two-Finger-Tap for Tablets

The gesture designed for tablets, namely Two-Finger-Tap, lets users zoom in by tapping the screen with two fingers suitably spaced. Figure 1 displays the whole zoom process: in the first step, users tap with two fingers the target area to be zoomed.

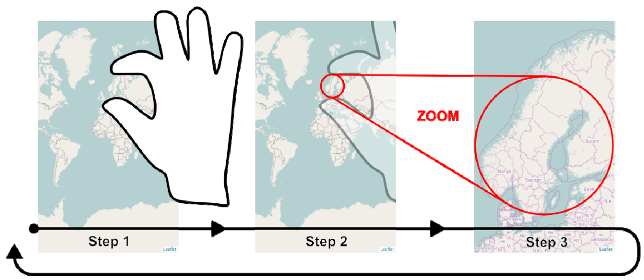

Figure 1. Two-Finger-Tap for Tablets. How it works.

In the second step, the algorithm identifies the area comprehended between the fingers (ideally, it may be a circle), which is enlarged automatically up to cover the map container as shown in step 3. The process is iterative: after the first zoom, the user can continue to zoom in starting again from the first step.

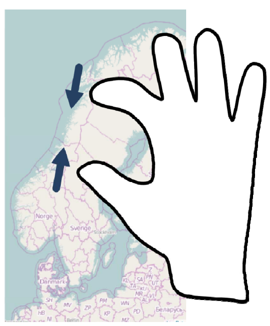

Figure 2. Two-Finger-Tap zoom out.

In order to zoom out, the traditional pinch-out has to be used as shown in Figure 2.

3.2 Tap&Tap for Smartphones

The gesture, namely Tap&Tap, lets users zoom in by touching in fast sequences two different points of the map (Figure 3, step 1).

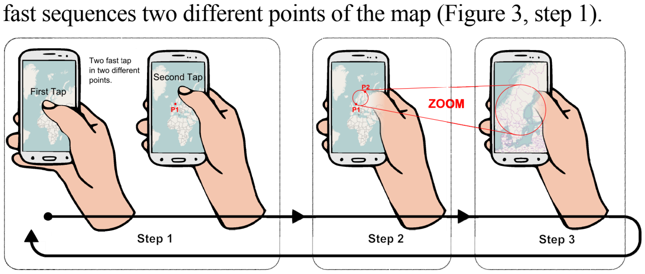

Figure 3. Tap&Tap for Smartphones. How it works.

The target area identified between the two points is enlarged automatically up to cover the map container (Figure 3, step 2 and 3).

Figure 4. Tap&Tap zoom out.

The one-hand usage is also kept to zoom out: users have to scroll the thumb from the left edge of the screen as shown in Figure 4.

3.3 Compatibility with Previous Gestures

The Two-Finger-Tap gesture comes into conflict with the gesture set of some commercial software for map navigation (e.g. Google Maps or Apple Maps): in their implementation, a tap with two fingers is used to zoom out one level in opposition to the traditional double tap (which is used to zoom in one level). At any rate, that gesture is unknown by users (none of the users who took the test was acquainted with it). Moreover, it seems not to be that useful because of the similarity to the traditional pinch-out (beyond the fact that two hands are needed in smartphones). Other frameworks for map navigation (e.g. Open Street Maps or Leaflet), in fact, do not implement that gesture. Moreover, neither Google mentions that gesture in the tips&trick official area[3]. For these reasons, that gesture is not taken into account in the following discussion. In order to understand the mechanisms through which the integration with the Pinch-Drag-Flick paradigm is designed without overlapping, a review of the traditional gestures is presented here:

1. Pinch: it allows the zoom adjustment by touching the screen with two fingers expanding/shrinking them (Figure 5, line A).
2. Drag: it allows the pan of the map by moving the finger in all directions (Figure 5, line B).
3. Double-Tap: it allows the zoom in with just one hand by using a double tap. The zoom starts from the point tapped (Figure 5, line C).
4. DoubleTap-and-Hold + Drag: it allows the zoom adjustment with just one hand by tapping twice, holding the screen the second time and moving the finger upwards (zoom out) or downwards (zoom in) (Figure 5, line D). This gesture was introduced by Google Maps.

The proposed gestures look like some of the aforementioned ones. However, they are quite different and their usage is not ambiguous. In particular, the Tap&Tap gesture for smartphones is similar to the Double-Tap. Anyway, the Tap&Tap occurs only when the second tap touches a different point from the previous one (Figure 5 line E), whereas the Double-Tap occurs only when the two taps touch the same point (Figure 5 line C).

Going on, the Two-Finger-Tap gesture is similar to the pinch. Anyway, the Two-Finger-Tap occurs only when fingers remain fixed on the screen and are released immediately after (Figure 5 line F), whereas the pinch occurs only when fingers move on the screen expanding or shrinking (Figure 5 line A, see touch move 1).

[3] https://support.google.com/gmm/answer/3273126

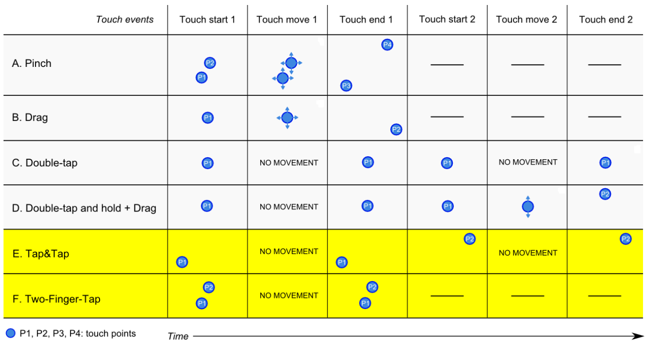

Figure 5. Recognition algorithm.
New gestures (in yellow) and traditional gestures.

The entire gesture set in Figure 5, without producing any overlap, lets users increase their possibilities to choose the most appropriate gesture in any conditions (tablets, smartphones, one-hand usage, two-hand usage). Finally, the conceptual differences between the pinch and the new gestures are clarified in order to better understand their nature. The pinch is "incremental": the zoom adjustment works continuously following the finger movements. The new gestures are "fit-area-to-container": the identified area comprehended between two points of the map is automatically enlarged up to cover the map container (like Adobe Acrobat X and Autodesk Autocad previously mentioned in the related work).

4. ON GESTURES IMPLEMENTATION

The new gestures were added to Leaflet, "*a JavaScript library for mobile-friendly maps*". The library exploits HTML5 and is compatible with any browser. The prototype designed – based on this library – works on smartphones and tablets without any difference. In particular, it was tested on a smartphone Motorola Moto G (4-inch, Android) and a tabletop Lenovo Flex 2-15 (15-inch, Windows 8.1). The browser was Google Chrome for both devices. In this paper, the exploitation of HTML technologies and how they may be applicable to develop novel interactions is put forward. The multi-touch management in HTML5/JavaScript is made possible by the events *touchstart*, *touchend*, and *touchmove*. These events were exploited to implement the new gestures (in addition to Pinch-Drag-Flick gesture set already offered by Leaflet). Few lines of code were required to redesign Leaflet. The most significant of them are discussed here.

```
1.   Before=0;
2.   TapAndTap=false;
3.   TwoFingerTap=false;
4.   $('#map').bind('touchstart', function(ev){
5.     After=Date.now();
6.     if (After-Before < 600) {
7.       TapAndTap=true;
8.       x1=ev.originalEvent.touches[0].clientX;
9.       y1=ev.originalEvent.touches[0].clientY;
10.    } else {
11.      TapAndTap=false;
12.      x0=ev.originalEvent.touches[0].clientX;
13.      y0=ev.originalEvent.touches[0].clientY;
14.      Before=Date.now();
15.    }
16.    if (ev.originalEvent.touches.length == 2)
17.      {
18.      TapAndTap=false;
19.      TwoFingerTap=true;
20.      x0=ev.originalEvent.touches[0].clientX;
21.      y0=ev.originalEvent.touches[0].clientY;
22.      x1=ev.originalEvent.touches[1].clientX;
23.      y1=ev.originalEvent.touches[1].clientY;
24.    }
25.  });
```

If two taps occur in fast sequence (up to 600ms of tolerance, line 6), the gesture Tap&Tap is enabled (line 7) and two screen points are identified (lines 8-9 and 12-13).

If two taps occur simultaneously (line 16), the gesture Two-Finger-Tap is enabled (line 19) and the two relative points are identified (lines 20-23).

```
26.  $('#map').bind('touchmove', function(){
27.    countMove++;
28.    if (countMove>4) {
29.      TapAndTap=false;
30.      TwoFingerTap=false;
31.    }
32.  });
```

If a movement of any finger occurs (line 28, the value 4 is the tolerance), the new gestures are disabled (lines 29-30) and the pinch or the drag is triggered by Leaflet, in its original implementation. Movements of the fingers on the screen, in fact, are not expected by the new gestures.

```
33.  $('#map').bind('touchend', function(){
34.    countMove=0;
35.    if (TwoFingerTap==true || TapAndTap==true) {
36.      DistanceBetweenPoints=CalcDBP(x0,x1,y0,y1);
37.      if (DistanceBetweenPoints > 24 ) {
38.        var point1 = L.point(x0, y0);
39.        var point2 = L.point(x1, y1);
40.        first=map.containerPointToLatLng(point1);
41.        second=map.containerPointToLatLng(point2);
42.        bounds = L.latLngBounds(second, first);
43.        distance=first.distanceTo(second)/2;
44.        circle=bounds.getCenter();
45.        newBnd=L.circle(circle,distance).getBounds();
46.        map.fitBounds(newBnd);
47.      }
48.    }
49.  });
```

When a finger releases the screen (line 33), if one of the two gestures is enabled (line 35), the distance between points x0, x1 and y0, y1 is calculated. If it is more than the tolerance value (set to 24, line 37), the new level of zoom is calculated and the map triggers the appropriate zoom automatically (line 46). If the distance between points is less than the tolerance value, the traditional double-tap, if it occurs, can be triggered (in the original implementation of Leaflet) zooming-in one level.

5. EVALUATIONS

Even though the paper is focused on the demonstrative aspects, a summary of the evaluations on the new gestures is displayed here. A more extended evaluation is planned for future work.

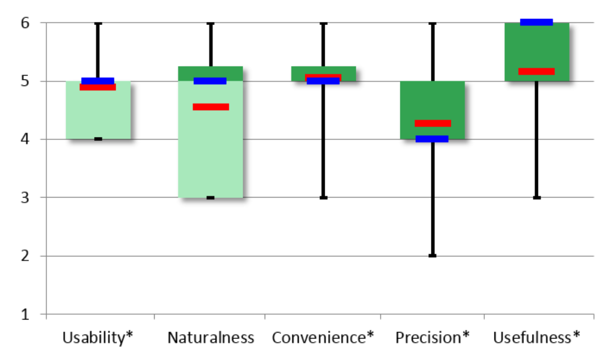

Figure 6. Tap&Tap evaluations on Smartphones.

Quantitative and qualitative evaluations were applied to 18 users in three steps: i) questionnaire with answers using a 6-point Likert scale, which evaluates different qualities (usefulness, usability, naturalness, convenience, and precision); ii) speed test to compare the execution time of the new gestures with the traditional pinch; iii) discussion with users (qualitative evaluation). In order to avoid bias, the speed test was applied after the questionnaire.

5.1 Questionnaire

A summary of the questionnaire results is displayed in Figure 6 for the Tap&Tap (smartphones) and in Figure 7 for the Two-Finger-Tap (tablets) through box-plots with mean (red) and mode (blue).

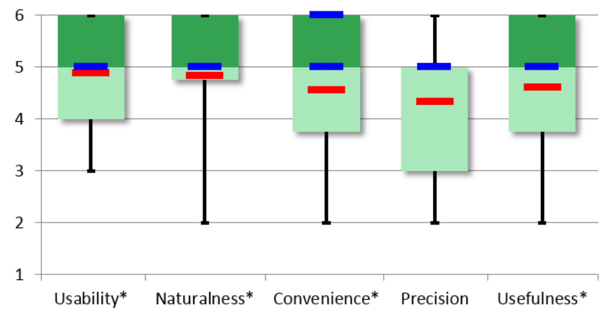

Figure 7. Two-Finger-Tap evaluations on Tablets.

The asterisks indicate clear positive tendencies ($p<0.05$) according to the binomial test (evaluation from 1 to 3 was considered as negative whereas from 4 to 6 was considered as positive).

5.2 Speed Tests

The speed test was carried out calculating the execution time of the new gestures in comparison with the traditional pinch. Four different zoom levels were evaluated: starting from the whole world, each user had to zoom in up to reach i) a part of Denmark (121820km2, lower zoom level), ii) Sicily, an Italian island (53122km2), iii) Berlin (2361km2), iv) Bicocca, a neighborhood of Milan (4,5km2, highest zoom level). The prototype calculated the execution times automatically for each zoom level. Table 1 displays the sum of the execution times on the four zoom levels (in milliseconds, average on 18 users).

Table 1. New gestures and Pinch time comparison. Time in milliseconds with standard deviation.

	Tablet Pinch	Tablet Two-Finger-Tap	Smartphone Pinch	Smartphone Tap&Tap
	Z=-3.680; p<0.001*		Z=-3.462; p=0.001*	
Total time	22186 (SD=4994)	15334 (SD=4995)	20482 (SD=2721)	17414 (SD=2595)

In both devices, the differences between the distributions are significant ($p<0.05$): Table 1 shows the test results according to the non-parametric Wilcoxon signed rank test. It was used because some of the data in Table 1 are not normally distributed (Shapiro-Wink test). 30% of time is saved on tablets whereas 14% of time is saved on smartphones. The execution times prove the good design qualities of the new gestures.

5.3 Qualitative Analysis

Due to spatial reasons, being exhaustive is quite difficult. At any rate, many users have shown real enthusiasm regarding the new gestures and, in particular, regarding the Tap&Tap (because of the one hand usage). Nevertheless, other users were not willing to change their regular practices. Anyway, when the users realized that the new gestures were added to the traditional pinch without replacing it, they perceived them as very positive.

6. Conclusions

All in all, both gestures were evaluated positively and can be implemented using Web technologies with few modifications: as shown, around 50 lines of code were enough to redesign Leaflet, the HTML5 framework used to develop the prototype presented in this paper. The implementation of the presented gestures in free and/or commercial software could be quite feasible.

7. Acknowledgment

Thanks to Daniela Bascuñan, a linguist who helped me to redraft this paper during a trip to Cuba among the sun, the beach and the sea. Actually, she was the translator of this paper. This work would not have been feasible without her. Thanks to Giorgio De Michelis, my PhD supervisor, for his support with his human, moral and academic qualities. Thanks for the trust he has placed and continues to place in me. Although we often disagree about ideas and concepts, undoubtedly, he is the main source of inspiration for the work I am doing in my PhD. Thanks to Flavio De Paoli, my other PhD supervisor, for the time employed to understand my bizarre ideas and me. Thanks also for the human support, besides the academic one. Thanks to Lorenzo Fusco, my old friend and physicist, for the time he devoted to review the statistical analyses giving me some pieces of advice.

8. REFERENCES

[1] Boring, S., Ledo, D., Chen, X. A., Marquardt, N., Tang, A., & Greenberg, S. (2012, September). 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* (pp. 39-48). ACM.

[2] Engholm, K. A., & Demory, M. D. (2003). *U.S. Patent No. 6,642,936*. Washington, DC: U.S. Patent and Trademark Office.

[3] Hinckley, K., & Song, H. (2011, May). Sensor synaesthesia: touch in motion, and motion in touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 801-810). ACM.

[4] Malacria, S., Lecolinet, E., & Guiard, Y. (2010, April). 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* (pp. 2615-2624). ACM.

[5] Spelmezan, D., Appert, C., Chapuis, O., & Pietriga, E. (2013, August). Side pressure for bidirectional navigation on small devices. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services* (pp. 11-20). ACM.

[6] Spindler, M., Schuessler, M., Martsch, M., & Dachselt, R. (2014, April). Pinch-drag-flick vs. spatial input: rethinking zoom & pan on mobile displays. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems* (pp. 1113-1122). ACM.

[7] Wigdor, D., Forlines, C., Baudisch, P., Barnwell, J., & Shen, C. (2007, October). Lucid touch: a see-through mobile device. In *Proceedings of the 20th annual ACM symposium on User interface software and technology* (pp. 269-278). ACM.