Interacting with Data Visualizations on Tablets and Phones: Developing Effective Touch-based Gestures and Operations

Ramik Sadana*

Georgia Institute of Technology

John Stasko

Georgia Institute of Technology

Abstract

Currently, data visualization systems employ a variety of techniques and user interface components to facilitate interaction on mouse-based desktop and laptop computers. However, the same is not true for interaction on touch-based devices such as tablets and mobile phones. Data visualization has yet to become pervasive there and it is not clear what interactive operations should be provided and how they should be performed. In this paper, we discuss an approach to designing interactions for touch-based visualizations employing user-suggested gestures. This design space for touch gestures on data visualizations is especially rich, perhaps more so than many expect.

Keywords: mobile and ubiquitous visualization, design studies

1 Introduction

Much of the power of data visualization systems comes from the flexible interactions they provide. Many systems include a host of interactive operations that support data exploration via selection, filtering, zooming, etc. Until now, data visualization systems have been developed primarily for desktop and laptop computers where the primary mode of interaction is through a keyboard and mouse. Today, however, with the proliferation of tablet computers and mobile phones, the primary interaction on such devices is touch by finger(s) or stylus. Accordingly, researchers and developers are now exploring ways to design visualization applications for tablets (examples include Tableau for iPad and Spotfire for iPad). To us, these applications still feel like desktop versions ported onto the touch screen, as they do not offer the smooth, optimized experience expected on tablets.

Touch-based interfaces provide a very different suite of potential operations, and current systems do not leverage this potential well yet. Nielsen [4] suggested that contemporary user interfaces should be replaced with gestural interfaces because interface elements such as buttons, menus, and status bars require significant screen real estate that may not be available on mobile displays. Drucker et al [2] explored different styles of interaction within data visualizations on tablets and showed that barchart interactions optimized for touch, e.g. by using gestures, were both faster for problem solving as well as more preferred compared to interactions based on traditional WIMP techniques.

Our objective in this research is to define an interaction grammar for visualization on touch-based tablet systems optimized across a variety of types of visualizations. We follow the approach of Morris et al [3] and utilize user-suggested gestures for interactions.

2 Related Work

A variety of past work examines touch interaction with visualization. However, most of the focus has either been on large touchscreen tables [7] or on specific visualizations such as node-link graphs [6]. For work particular to tablets, Baur et al [1] examined the stacked graph for their TouchWave system, and developed a new set of interactions. They created a variety of multi-touch gestures for scaling, scrolling, providing context, extracting layers, and many other activities. They noted that developing a consistent interaction set was one of the primary challenges of the work.

Recent work by Rzeszotarski and Kittur [5] describes TouchViz, a visualization system for tablets that leverages the direct manipulation of fingers to interact with data. The researchers employ magnifying glass and zoom lens interactions, stressing the playfulness of the interface using physics and gravity. However, they do not discuss how the new interactions affect the efficiency of problem solving or how the interactions would scale for multiple visualization types.

3 Designing Touch Interactions with Users

Although our goal is to establish an interaction grammar across multiple types of visualizations, we decided to begin with one specific type of visualization for initial exploration. We selected the scatterplot, one of the fundamental techniques for depicting low-dimensional multivariate data. To understand the interaction operations possible on scatterplots, we reviewed a number of existing systems that employ them, such as Spotfire, Tableau, and Microsoft Excel. Our aim was to identify a broad set of possible operations that can be performed on a scatterplot. Our review generated a list of around 30 different operations, which we further grouped under the visualization task/intent hierarchy provided by Yi et al [8]. Examples included *Encode* (assign variables to the x and y axes, assign variables to mark color, shape or size), *Filter* (on any axis or non-axis variable), and so on.

We used this list to conceptualize a formative user study whose aim was to identify a set of gestures that participants would suggest for the various operations. In the study, we gave participants an iPad showing an image of a scatterplot along with a description of the dataset being represented. This description included details of variables that were not currently represented as well. The image on the iPad was static and unresponsive to any gestures the user performed. Prior to the study, we generated a set of ten questions based on the dataset. We designed the questions to collectively cover all the operations mentioned. For example, one question asked participants how they would change the variable mapped to the x-axis.

In the study, participants were given one question at a time and asked how they might solve it or perform the task using any type of gesture or interactive operation. After each gesture, we asked the participants to describe how the screen would appear. We also pushed the participants to suggest different possible gestures for each question. For example, for the question regarding how to change the variable on the x-axis, Participant 2 mentioned tapping on the x-axis to reveal a menu, or swiping in from the right to reveal a list of all variables. The study included 7 participants and generated a list, on average, of about 6 different gestures for each

*email:{ramik@gatech.edu, stasko@gatech.edu}

operation. Table 1 presents a snapshot of the set of user-suggested gestures (along with frequency) for the Assign Axis task.

We made a number of observations at this stage. First, there was significant variation in participants' preferences. Participants routinely suggested very different gestures for the same task, as well as associated the same gestures for very different tasks. Second, we found that a particular participant would frequently use the same gesture for different actions at different stages of the study. Finally, participants often employed gestures that they had experience with in the past in some other user interface.

Table 1. List of user-suggested gestures for the task ‘Assign Axis’

Assign Axis	
Tap or double tap on axis to open a list.	4
Tap or double tap on the label to open a list	2
Button on top right to open a menu – drag variable on axes	5
Slide open menu from right – drag variable on axes	1
Draw the variable name on the axis	1
Auto-assigned variables as default view	1
Variables aligned at the top horizontally	2

3.1 Finalizing gestures

In the second phase of the user study, we built on the results of the first phase by prototyping some of the gestures that participants had suggested. We used an Apple iPad, programming the interactions in Objective-C using the native graphics library provided by iOS. To begin, we focused on two tasks – selection and zooming. For each, we implemented the top three preferred gestures.

As an example, in the new prototype of a scatterplot, we supported zooming in one of three ways:

1. Two finger pinch in or out on an axis-line to zoom into that axis.
2. Tap and drag on an axis to create a marquee over the dragged region. Then double tap on the marquee to zoom into the selected region.
3. Double tap on the visualization to zoom in, with the touch point as the centroid. The visualization zooms in to minimize the number of overlaps in the visible set of points.

Our initial aim was to go back to the participants with a working prototype and evaluate their inclinations again. We believed that performing multiple rounds of iterative refinement with users would eventually generate a clear set of results. However, preliminary user testing of the hi-fidelity prototype revealed that finding such singular preferences is not straightforward. Unlike experiences with buttons and menus, there is a large variation in the way users experience gestures as a means of interaction. One of the main reasons for this stems from the fact that gestures are a learned behavior that tends to improve for users over time.

Even within our own testing of the interface, we realized that it was not easy to choose one gesture over another for a particular operation. Learnability of the gesture is important, but so is the discoverability of the gesture across different types of visualizations. These factors, however, were not affecting the decisions of the participants when they chose one gesture over another. Furthermore, at this early stage of design, it is difficult to predict conflicts that may arise during prototyping of gestures for other visualization types.

4 Discussion

Reflecting on our approach of designing interactions for the scatterplot via user-suggested gestures, we identified that the inherent weakness of this approach was the lack of novelty of gestures. Participants in the study had varying amounts of experience using touch-based interfaces. It was notable that all the gestures that the participants suggested were ones they had some past experience using. While describing the gestures, participants often referred to an application they had used on their phone or tablet that employed that gesture.

We feel that this does not help our cause, since on touch devices data visualization systems offer a design challenge different than most others. The primary reason for the difference is the rich variety of operations that need to be supported. Most, if not all, are actions that do not have a substitute. For instance, filtering as an operation is very different than sorting or selection. Given such a large set of tasks, we feel that user suggestions did not reveal a rich enough set of gestures to cover all of it.

There are very few other types of applications that face similar issues. Some are applications such as Adobe Photoshop or Microsoft Office that have a very rich interface on desktop machines, with a wide array of tools and operations. Porting these applications to tablets that do not offer enough real estate to make all options available is difficult. As a result, even the most recent iterations of these applications offer a very restricted set of features, which likely limits users from switching to tablets for all their activities.

We feel that providing visualization systems with limited feature sets would not be a good approach. Although visualization interfaces typically are not as feature rich as Photoshop or Word, any visualization application that supports data exploration and problem solving needs rich functionality available in order to be useful. Hence, the answer lies in finding the right suite of gestures, balancing both the functionality of the system as well as the usability and learnability of the gestures. As we continue to explore the variety of touch interactions, we believe that a resulting suite of useful gestures will be a combination of user-suggested gestures, existing gestures from other application domains, and the novel gestures developed through new research.

References

[1] D. Baur, B. Lee, and S. Carpendale. TouchWave: Kinetic Multi-touch Manipulation for Hierarchical Stacked Graphs. *Proceedings of ITS ‘12,* pages 255-264, Nov. 2012.

[2] S.M. Drucker, D. Fisher, R. Sadana, J. Herron, and m.c. schraefel. TouchViz: a case study comparing two interfaces for data analytics on tablets. *Proceedings of ACM CHI ’13*, ACM, April 2013.

[3] M. R. Morris, J.O. Wobbrock, and A.D. Wilson, Understanding users' preferences for surface gestures. *Proceedings of Graphics Interface,* pages 261-268, June 2010.

[4] J. Nielsen. Jakob Nielsen’s Alertbox: Browser and GUI Chrome. http://www.useit.com/alertbox/ui- chrome.html, 2012.

[5] J.M. Rzeszotarski, A. Kittur. TouchViz: (multi)touching multivariate data. *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 1770-1784, ACM, April 2013.

[6] S. Schmidt, M. Nacenta, R. Dachselt, S.A. Carpendale. A Set of Multitouch Graph Interaction Techniques. *Proceedings of ITS ‘10*, pages 113–116, Nov. 2010.

[7] S. Voida, M. Tobiasz, J. Stromer, P. Isenberg and S. Carpendale. Getting Practical with Interactive Tabletop Displays: Designing for Dense Data, ‘Fat Fingers,’ Diverse Interactions, and Face-to-Face Collaboration. *Proceedings of ITS ‘09*, pages 109–116, Nov. 2009.

[8] J.S. Yi, Y.A. Kang, J. Stasko, J. Jacko. Toward a deeper understanding of the role of interaction in information visualization. *IEEE Trans. Visualization and Computer Graphics,* 13: 1224-1231, 2007.