TouchPivot: Blending WIMP & Post-WIMP Interfaces for Data Exploration on Tablet Devices

Jaemin Jo[1] **Sehi L'Yi**[1] **Bongshin Lee**[2] **Jinwook Seo**[1]
[1]Seoul National University, Seoul, Republic of Korea
[2]Microsoft Research, Redmond, WA, USA
{jmjo, shlyi}@hcil.snu.ac.kr, bongshin@microsoft.com, jseo@snu.ac.kr

ABSTRACT

Recent advancements in tablet technology pose a great opportunity for information visualization to expand its horizons beyond desktops. In this paper, we present TouchPivot, a novel interface that assists visual data exploration on tablet devices. With novices in mind, TouchPivot supports data transformations, such as pivoting and filtering, with simple pen and touch interactions, and facilitates understanding of the transformations through tight coupling between a data table and visualization. We bring in WIMP interfaces to TouchPivot, leveraging their familiarity and accessibility to novices. We report on a user study conducted to compare TouchPivot with two commercial interfaces, Tableau and Microsoft Excel's PivotTable. Our results show that novices not only answered data-driven questions faster, but also created a larger number of meaningful charts during freeform exploration with TouchPivot than others. Finally, we discuss the main hurdles novices encountered during our study and possible remedies for them.

Author Keywords

Information visualization; data exploration; pen and touch interaction; natural interaction; table devices; pivot; novices.

ACM Classification Keywords

H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces.

INTRODUCTION

Information visualization (InfoVis) is expanding its horizons to a wider audience. People are transforming their data into visualizations for sensemaking, decision making, storytelling, and casual purposes [12]. The increasing demand for visualizations has spawned many commercial visualization tools (e.g., Tableau [30], Qlik [22]) as well as web-based systems (e.g., iVisDesigner [23], Voyager [35]). Most of these tools enable people to transform their data and visualize it in the form of familiar charts such as bar charts and line charts.

Meanwhile, new interaction modalities such as pen and touch interactions have been emerging and spreading out with the proliferation of tablet devices. The rise of new interaction modalities opened up a new possibility in designing interactive visualization systems. For example, several pioneering studies and products have shown how these new interactions could be used for various charts such as bar charts [5, 31], line charts [31], scatterplots [24, 25], stacked graphs [1], and wordles [14]. In this paper, we broaden the possibility of exploratory data analyses on tablet devices by investigating pen and touch interactions for supporting common yet essential data transformations (e.g., pivot transformations).

On desktop environments, we can find interfaces that allow people to transform tabular data and visualize the results. Popular and widely-used interfaces include PivotTable in Microsoft Excel [20] and Tableau [30]. They take a similar approach, called *shelf configuration*, for data transformations and visualization construction. In that scheme, people can set the roles of columns by dragging-and-dropping column names to one of the "shelves." For example, PivotTable provides four shelves: *Rows*, *Columns*, *Values*, and *Filters*. People can move a categorical column (e.g., *gender*) to *Rows*, and a numerical one (e.g., *age*) to *Values* to aggregate it over the categorical one (i.e., the average age of men and women) and visualize the result through a bar chart.

The main strength of the shelf configuration scheme is the flexibility in configuration, providing a large degree of freedom in tweaking data transformations and visualizations. However, as a previous study [9] points out, novices who are not familiar with such data transformations and visual mappings have trouble in exploiting such flexibility. To help people including novices make the best use of the power of visual data exploration, we set out to design more natural and familiar user interface and interactions for data exploration, considering their hurdles.

To tackle these challenges, we designed TouchPivot (Figure 1), a novel interface that assists visual data exploration on tablet devices by leveraging pen and touch interactions. Keeping novices in mind, we seek to support simple and

CHI 2017, May 06 - 11, 2017, Denver, CO, USA

ACM 978-1-4503-4655-9/17/05$15.00
DOI: http://dx.doi.org/10.1145/3025453.3025752

rapid data exploration rather than providing full flexibility in configuration of transformation and visualization. We conducted a user study with 48 novices to compare TouchPivot with Tableau and PivotTable. Our study results showed that participants not only answered data-driven questions faster, but also created a larger number of meaningful charts during freeform exploration with TouchPivot than others. We also observed the main hurdles people encountered during our study. Finally, we discuss possible remedies for them, and consider potential limitations and future work.

RELATED WORK

In this section, we briefly describe three relevant topics: 1) previous work designing pen and touch interaction, 2) the shelf configuration scheme and interfaces that employed the scheme, and 3) studies investigating how novice people construct and understand visualizations.

Pen and Touch Interaction

Recognizing the great potential provided by the advances in interaction technologies, several researchers have explored the possibilities of pen and touch interactions. For example, Hinckley et al. [11] explored the opportunity of multimodal pen+touch inputs in designing new interfaces. In addition, visualization researchers discussed the challenges and research opportunities in supporting interactive visualization on touch-enabled devices [13] or going beyond the traditional WIMP (windows, icons, menus, pointers) environment [15]. A group of studies aimed at leveraging pen and touch interaction on interactive surfaces for various activities, such as diagram editing [7], graph exploration [28], data analysis [3, 4, 17, 33, 36], and storytelling [16]. Another set of research has focused on designing pen and touch interactions for visualization on tablet devices. Popular visualization and interaction techniques were redesigned on tablet devices, such as bar charts [5, 31], line charts [31], scatterplots [24, 25], stacked graphs [1], wordles [14], multiple coordinated views [26], and data selection [27]. TouchPivot contributes to this line of research by designing a data exploration interface for tabular datasets that novices can easily learn and use.

To elicit a gesture set that novices may easily understand and use, we surveyed pen and touch gestures used in 13 studies that enabled data exploration with pen and touch interactions [1, 3, 4, 5, 14, 16, 17, 24, 25, 26, 31, 33, 36]. Table 1 shows the top five most common gestures for each input modality. To keep our gesture vocabulary as small and simple as possible, we decided to use three touch gestures (i.e., tap, press-and-hold, and drag) and four pen gestures (i.e., tap, simple stroke, lasso, and writing a letter). We also surveyed pen and touch gestures for a specific action (e.g., selection, filtering), and reflected the result in our design. In the Design Process section, we elaborate more on our design process.

Shelf Configuration Interfaces

Grammel et al. [8] classified visualization construction user interfaces into six categories. Among them, shelf configuration is one of the most widely used types of interfaces in existing tools. The shelf configuration interfaces provide a fixed number of shelves and people can apply data transformations or create visual mappings by dragging-and-dropping columns into the shelves. Following the shelf configuration scheme, Microsoft Excel [20] enables people to generate pivot tables and pivot charts through dedicated interfaces named PivotTable and PivotChart. Tableau [30] is another powerful visualization tool that also exploits the shelf configuration scheme. In contrast to PivotTable, Tableau hides data tables from people so that people can focus on creating visual mappings between data attributes and visual properties. However, hiding data tables can confuse novices who are not good at conceiving data transformations, as revealed in our user study. Voyager [35] facilitates data exploration by recommending effective visualizations, extending the shelf configuration scheme. Similarly, while not employing the shelf configuration scheme, TouchPivot suggests useful visual mappings to support rapid exploration of legitimate data transformations and visualizations through pen and touch interactions.

Touch Interaction (12 studies)	Pen Interaction (6 studies)
Drag (11)	Simple strokes[a] (6)
Tap (10)	Lasso (5)
Pinch (8)	Writing a letter (4)
Press-and-hold (5)	Tap (3)
Lasso, Double tap (4)	Scribbles (2)

Table 1. Top five most common gestures for each input modality in previous studies. Numbers in parentheses indicate the number of studies that used each gesture. [a]Simple strokes include single straight strokes, such as a strikethrough.

Information Visualization for Novices

Novices have been regarded as a new audience for InfoVis. Starting to pay more attention to novices, researchers have conducted studies specifically investigating how InfoVis novices construct and understand visualizations. Lee et al. [18] presented a grounded model that described how lay people make sense of unfamiliar visualizations. Grammel et al. [9] identified three challenges that the novices have difficulties with: data selection barrier, visual mapping barrier, and interpretation barrier. Following their design implications for novices, TouchPivot suggests visualizations and supports iterative specification. Heer et al. [10] characterized novices in terms of skills, goals, and datasets, asking for tools that help novices easily create and deploy visualizations. We believe TouchPivot can be an example of such tools. Elias and Bezerianos designed Exploration Views (EV) [6] that helped novices easily build information dashboards. We also reflected on their design guidelines for novices into our design such as highlighting linked selections on a table and visualizations without affecting the visual representation of unselected data.

TOUCHPIVOT

Design Rationale

TouchPivot is designed to enable simple and rapid visual data exploration on tablet devices. We iteratively refined our design, keeping the following design rationale in mind.

Seek for simplicity rather than flexibility and power

In contrast to most pivot interfaces available on desktops, which provide people with a full control of data transformations, we seek to provide lightweight key features with simple interactions. The reasons behind this design decision are that 1) novices tend to underutilize such full control [9], 2) rich features and interactions on tablet devices are often undiscoverable and thus should be remembered, and 3) we can cover important common cases without such powerful features.

For example, PivotTable allows people to pivot data using any columns regardless of their types (i.e., categorical or numerical). However, pivoting the data by numerical attributes often yields undesirable results (e.g., too many values on the x-axis), which may confuse novices. Although there might be a specific situation where people do such pivoting deliberately, we observed such freedom tended to make the interface demanding and error-prone. In TouchPivot, people do not have to designate the role of a selected column, but just select columns that they want to involve in a pivot transformation. TouchPivot heuristically decides which data transformations are appropriate depending on the types of the chosen columns (Table 2).

Leverage pen and touch interaction

Previous studies [1, 5, 14, 15, 17, 24, 25, 36] have shown that pen and touch interactions are promising alternatives to keyboard and mouse interactions. Inspired by these encouraging results, we design pen and touch interactions for data exploration on table devices, based on a gesture set elicited from previous studies. Note that the goal of this paper is not to present novel pen and touch gestures for data exploration, but to design a data exploration interface that novices can easily learn and use. We detail our pen and touch interactions in the Interacting with TouchPivot section.

Column Types	Histogram	Bar Chart	Line Chart	Scatter-plot	Pivot Table
N[a]	O				O
N_1, N_2[b]		O		O	
C	O				O
C, N		O	O		O
C, N_1, N_2		O	O	O	O
C_1, C_2	O				O
C_1, C_2, N		O	O		O
Other cases					O

Table 2. Suggested visualizations according to the selected column types. For example, the row with "C_1, C_2, N" lists suggested visualizations when two categorical columns (i.e., C_1 and C_2) and a numerical column (i.e., N) are selected. The order of columns does not matter. For example, TouchPivot shows the same visualizations for "C, N" and "N, C." [a]A box plot and descriptive statistics (e.g., quartiles) were provided. [b]Correlational statistics (e.g., Pearson coefficient) was provided.

In shelf configuration interfaces, people have to drag-and-drop every column once to explore data, and this tedious repetition could hinder novices' data exploration. To facilitate quicker and easier data exploration, we designed the *fan menu* on the bottom left of the screen (Figure 1a) inspired by bimanual interaction [32]. People can move their left thumb on a wedge-shaped area to rapidly pivot the data with every column one by one and check the results through suggested visualizations.

Suggest useful visual mappings

Visualizing transformed data is an essential step in data exploration. However, as Grammel et al.'s study [9] demonstrated, people who are not familiar with InfoVis have difficulties in designing visual mappings (i.e., the visual mapping barrier). As a remedy, we decided to suggest useful visual mappings and possible visualizations for every pivot transformation. We enumerated potential visual mappings for each combination of column types in a pivot transformation and evaluated them based on its effectiveness and familiarity. Note that we also considered familiarity as a criterion because people preferred familiar visualizations in the previous study [9]. Table 2 shows our rules for visualization recommendation. With a simple swipe gesture on a visualization, people can quickly go through alternative visual mappings (e.g., a line chart instead of a bar chart).

Utilize a connection between data tables and visualizations

In Tableau and PivotTable, people can interact only with an abstract form of data instead of directly manipulating or seeing the raw data. For example, Tableau represents the data using column names in its workspace, not showing all rows and values of the data. While this can be a succinct summarization of data for effective exploration, we believe such abstraction imposes a cognitive burden on people, because they have to interpret data transformations indirectly (for example, by conceiving the transformed data from visualization). In TouchPivot, we show the transformed data (or raw data when no data transformation was applied) in its original form (i.e., a table) on the half of the screen juxtaposed with suggested visualizations (Figure 1).

In addition to showing a data table, we tightly couple the data table and visualizations together. To convey a connection between them, TouchPivot employs a consistent color encoding, a well-known method to relate information [34] (Figure 1b). Additionally, TouchPivot keeps the data table and charts coordinated throughout all interaction including data-related interaction (e.g., filtering, sorting, and selection) and chart-related interaction (e.g., chart type change).

Design Process

As mentioned in the Related Work section, we first reviewed previous pen and touch data exploration interfaces and elicited a simple gesture set from them. Our gesture set included three touch gestures (i.e., tap, press-and-hold, and drag) and four pen gestures (i.e., tap, simple strokes, lasso, and writing a letter).

To find an effective mapping between gestures and actions, we also surveyed gestures mapped to common actions in data exploration, such as filtering. We identified several actions that were consistently mapped to the same gesture. For example, five out of six studies used the simple stroke gesture for filtering out data. Similarly, in four out of six studies, writing a letter was mapped to changing aggregation functions (e.g., SUM to MAX). For these actions, we could bring in the mapping to our design. However, we also found actions that were inconsistently mapped. For example, among 12 studies that used touch interaction, three interfaces employed the tap gesture for filtering, while other three interfaces used the drag gesture for the same action. We also found that another two interfaces supported both. For this case, we first incorporated both gestures in our initial design and chose a better gesture based on pilot studies.

With the initial design, we performed two pilot studies with 31 participants in total, using TouchPivot and Microsoft Excel's PivotTable. All participants had never used both TouchPivot and PivotTable, and they used both interfaces (i.e., a within-subjects design) for two hours (i.e., one hour each). After watching a 15 minute-long tutorial video, we asked participants to try the two interfaces to answer several data-driven questions (Table 3) and freely explore given datasets for five minutes. We measured the time and accuracy of answers, collected qualitative feedback, and identified major usability issues.

Through two pilot studies, we learned that novices prefer to use traditional WIMP-style widgets for specific tasks. For example, they found filtering widgets (i.e., checkbox lists) in PivotTable were familiar and helpful in identifying filtered values, though they could remember our strikethrough interaction for filtering out items with a pen. We thus decided to incorporate two WIMP widgets (i.e., the filter view on the right side of the screen in Figure 1a and the drop-down lists on the title of charts in Figure 2b) in our design. We also removed several interactions that confused novices. For example, TouchPivot initially enabled novices to swipe down a bar in a bar chart to filter it out. However, we removed this interaction because people often made a mistake in differentiating it from the undo gesture (i.e., swiping down a chart).

Interacting with TouchPivot

We describe TouchPivot's interactions by following a simple scenario where an educational researcher, Emma, first explores the performance of her students. The data contain information of 200 students who took the same course. The data consists of six columns: *student ID*, *gender*, parents' education levels (*Medu* and *Fedu* for mother's and father's, respectively), and two numerical scores (*midterm* and *final*). Please refer to our supplementary video to see TouchPivot's pen and touch interactions.

Main Interface. The main interface of TouchPivot consists of four parts: a data table on the left side, a chart view in the top right, a chart preview in the bottom right, and a WIMP-style filter view on the far right side (Figure 1a). Data are shown in both a tabular form (i.e., the data table) and visualizations (i.e., the chart view).

Initialization. Initially, TouchPivot shows the data in the table. Emma confirms that rows and columns are loaded successfully. Then, she sets aside a less interesting column in her analysis, *student ID*, by performing a swipe-right gesture on the column header.

Exploration with the Fan Menu. As the first step of exploration, Emma wants to see the distribution of values on each column. She holds her left thumb on the bottom left edge of a fan menu (Figure 1a). Then, the first column (i.e., gender) is highlighted and the distribution of values on the column is visualized with a bar chart in the chart preview. She checks how many males and females took the course. As she gradually drags her thumb to the bottom right on the fan menu, the corresponding column in the data is highlighted in turn. Through this simple touch interaction on the fan menu, Emma can quickly explore the results that are obtained by

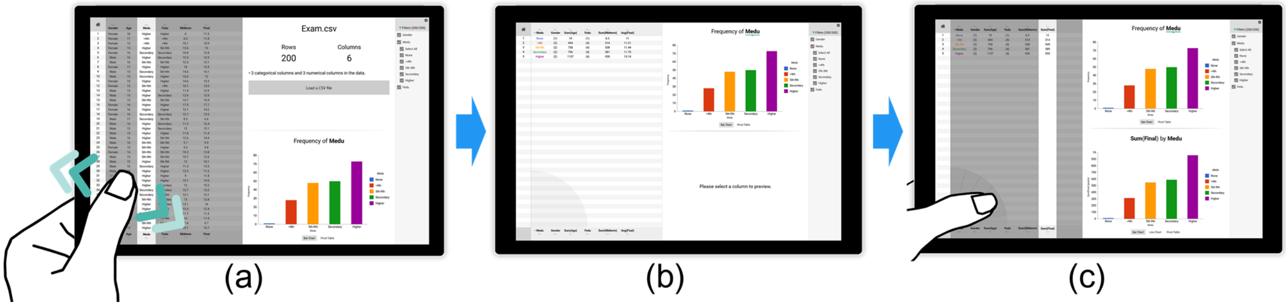


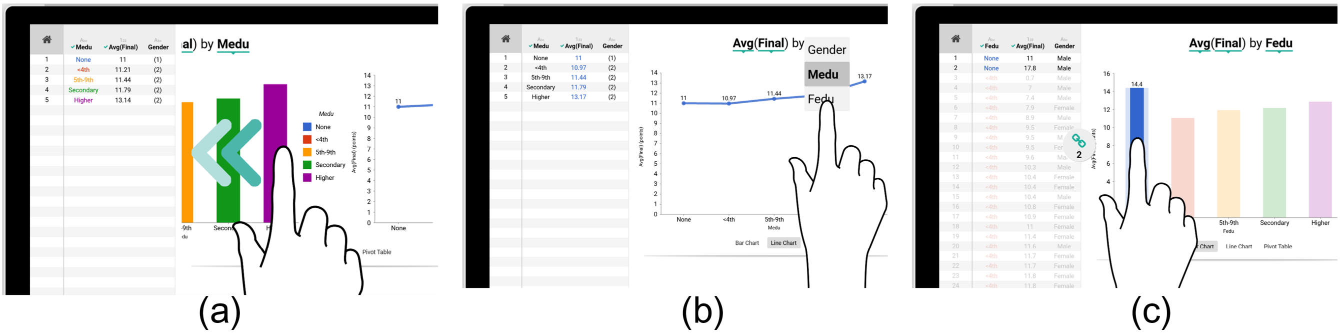
Figure 1. Visual data exploration using a fan menu. (a) A researcher, Emma, is exploring her data using the fan menu. When she taps on the fan menu which is on the bottom left of the data table, the corresponding column (e.g., *Medu*, mother's education level) is highlighted. The chart preview is visualizing the frequency of each category in the *Medu* column using a bar chart. She can drag her thumb to see the distribution of other columns. (b) When she lifted her thumb, the bar chart moves up to the chart view and the data are pivoted by the selected column (i.e., *Medu*). She can undo the pivot transformation by pulling down the bar chart to the chart preview. (c) After pivoting the data with the *Medu* column, she can continue her exploration by selecting another column.

pivoting the data with each categorical column or binning the data with each numerical column.

Pivoting the Data. Using the fan menu, Emma finds a categorical column, *Medu* (*mother's education level*), which looks like an interesting independent variable. To select the chart, she lifts her left thumb from the fan menu when the *Medu* column is highlighted (Figure 1a). When her thumb lifted, two things happen (Figure 1b). First, the table, which previously showed the raw data, changes to show the data pivoted by the *Medu* column. Emma sees rows with the same value on the target column gather up together and being merged into a single aggregated row through an animation. Secondly, the bar chart on the bottom right (on the chart preview) moves up to the top (the chart view), visualizing the pivoted data. Emma now wants to check whether *Medu* is related to final score. She directly taps on the header of the *final* column in the data table to select it. A new bar chart is created and stacked in the chart view, visualizing the sum of the final scores over mother's education level.

Tweaking the Visualization. Emma now wants to check the average of final scores. She starts writing an "a" (for average) with a pen on the header of the *final* column. The bar chart changes to show the average final scores for each level of *Medu*. Looking at the bar chart, she notices that the average of students' final scores increases as their mothers' education level increases. Then, Emma wants to see the increasing pattern in a more salient form. She turns over available charts through a swipe gesture (Figure 2a) and chooses a line chart. Now, she is curious about whether a similar trend also exists for father's education level. She taps on *Medu* on the title of the chart, and a WIMP-style drop-down list appears, showing all other categorical columns that can replace the current one (Figure 2b). She taps on *Fedu* (father's education level) and the chart changes to show the average final scores over father's education level.

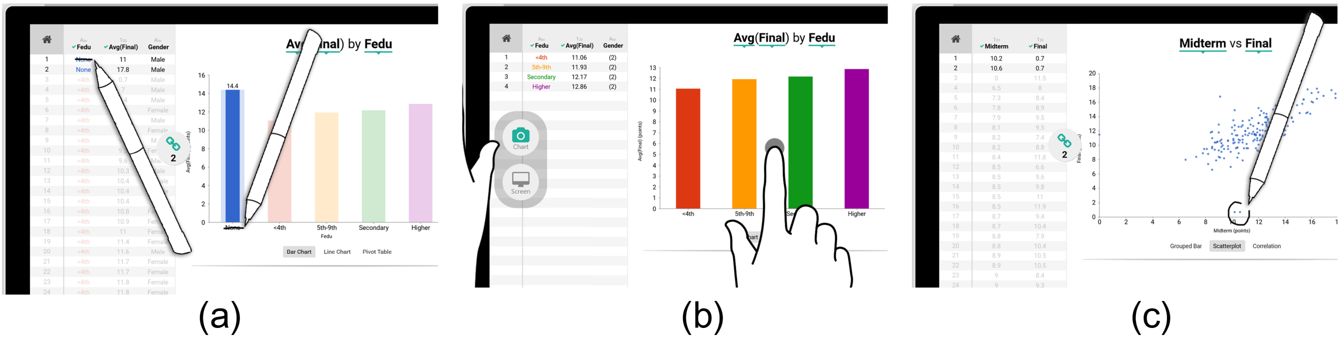
Selection and Filtering. Emma confirms that a similar trend also exists for father's education level. However, the line chart shows a spike when father's education level is "None." To see the details about such an outlying pattern, she turns back to a bar chart and taps on the bar that corresponds to "None" (Figure 2c). Then, other bars become unhighlighted, and raw records that correspond to the selected bar (i.e., rows whose father's education level is "None") appear on the table. Emma notices there are only two records, and those records made the spike. She strikes through the "None" value on the table using a pen (Figure 3a). The outlying records become grayed out, indicating they have been filtered out. In the filter view, the checkbox next to "None" under father's education level becomes unchecked. To capture the chart into the clipboard, she holds her right index finger on the chart, and taps on a camera icon that appeared on the left edge of the screen with her left thumb (Figure 3b). Now, she pastes the chart into a digital notebook for later use.



	Medu	Avg(Final)	Gender
1	None	11	(1)
2	<4th	11.21	(2)
3	5th-9th	11.44	(2)
4	Secondary	11.79	(2)
5	Higher	13.14	(2)

	Medu	Avg(Final)	Gender
1	None	11	(1)
2	<4th	10.97	(2)
3	5th-9th	11.44	(2)
4	Secondary	11.79	(2)
5	Higher	13.17	(2)

Figure 2. Touch interactions in TouchPivot. (a) People can explore an alternative visual mapping (i.e., a line chart) for a bar chart through a swipe gesture. (b) People can change the columns in a pivot transformation by tapping on the names in a chart title and selecting other columns. (c) TouchPivot keeps a data table and visualizations in sync. When people tapped on a bar in a bar chart, the raw records belong to the bar are selected and shown in the table. People can tap on the bar again to unselect the records.



	Medu	Avg(Final)	Gender
1	<4th	11.06	(2)
2	5th-9th	11.93	(2)
3	Secondary	12.17	(2)
4	Higher	12.86	(2)

Figure 3. Pen and touch interactions in TouchPivot. (a) People can filter out a category by striking out the category on the table or visualizations. (b) To capture a chart into the clipboard, people can press-and-hold a finger on a chart, and tap on a camera icon on the widget appearing from the left edge of the screen. (c) People can draw a lasso on a scatterplot to see details on the points contained in the lasso. Similarly, they can draw a circle on the table to locate the selected records on the scatterplot.

Creating a Scatterplot. Emma wants to check if there is a correlation between midterm and final scores. She taps on the home button on the top left corner of the screen to initialize the interface; TouchPivot shows the initial screen, resetting all column selections. Emma taps on the headers of the *midterm* and *final* columns on the table. TouchPivot first shows a bar chart that compares the sums of two scores. As she turns over the bar chart through a swipe-left gesture, a scatterplot appears, showing the relationship between the two scores (Figure 3c). She notices there is a positive correlation between the two scores, but there are also some outliers. She draws a lasso using a pen around those outliers to see the details about them on the table. Then, on the *midterm* column in the table, she draws a vertical stroke from top to bottom to sort the selected rows by the midterm score in descending order. She continues her analysis, examining other attributes of the outlying records.

USER STUDY

Pivot Interfaces

We compared TouchPivot with the pivot interfaces of two widely-used commercial products, Microsoft Excel 2016 (PivotTable) and Tableau 9.3. Both interfaces were designed based on the shelf configuration scheme; they provided several shelves where people could drag and drop column names to designate the roles of the columns in data transformations and visualizations. In PivotTable, people could add a categorical column to the *Rows* shelf to pivot data by the column. Similarly, they can add a numerical column to the *Values* shelf to aggregate the column using an aggregate function (e.g., average, sum, min, and max). Tableau called similar features with different names, *Columns* instead of *Rows* in PivotTable and *Rows* instead of *Values* in PivotTable.

They also provided different degrees of freedom in visualization construction. PivotTable only provided four roles for a column (i.e., *Rows*, *Columns*, *Values*, and *Filter*) while Tableau supported more flexible visualization configuration by providing more roles such as *Rows*, *Columns*, *Color*, *Size*, and *Label*. In addition, PivotTable did not distinguish the column types, while Tableau showed columns in separated lists according to their types; *Dimensions* for categorical columns and *Measures* for numerical columns.

Apparatus

All interfaces were run on the same tablet device, Microsoft Surface Pro 3 (hereafter *Interface*), which supported multi-touch and pen interactions. The Interface was equipped with a 12-inch 2160x1440 pixel screen. We increased the zoom factor of the screen to 125% to prevent menus and buttons from becoming too small. We placed the device on an integrated stand. In addition to touch, participants could use a pen (a Surface Pen) to interact with TouchPivot. For other interfaces (i.e., PivotTable and Tableau), we had participants use a keyboard and a mouse which were the input modality that the interfaces were designed for. We preconfigured Interface before each session to minimize the effect of external factors such as time and effort to load or set up data. Participants received a preloaded instance of the given interface with the data loaded, and thus they could start all tasks instantly without any extra setup.

In addition to Interface, we laid another tablet device (hereafter *Presenter*), 9.7-inch iPad, on the table between a participant and Interface. *Presenter* ran a session manager (a web app) to present questions, receive answers from the participants, and measure task completion time.

Participants and Study Design

We recruited 48 participants (33 males and 15 females) from a university, ages 18 to 28 years. All participants were right-handed and had never used any of the three interfaces. We also screened participants for color blindness and required normal or corrected-to-normal eyesight. They received about US$10 for their participation.

We used a between-subjects design with three interface conditions: TouchPivot, Tableau, and PivotTable. Sixteen participants were assigned to each interface condition.

Tasks and Datasets

We designed two tasks that represent different use cases: Task 1 for targeted analysis (i.e., answering a specific data-driven question) and Task 2 for freeform exploration.

For Task 1, we derived six questions on a movie dataset (Table 3) and asked participants to find an answer for each question as fast and accurately as they could. The questions had two difficulty levels: three questions (i.e., easy questions, Q1-Q3) involved two columns, and the other three questions (i.e., hard questions, Q4-Q6) involved three or four columns. Four questions (Q1, Q2, Q4, and Q5) asked for a specific value (i.e., a number or a category name). In

Name	Difficulty	Question Description
Q1	Easy	Grouping movies according to their *age* and *MPAA rating*, how many movies are in the largest group?
Q2		Of movies whose *creative type* is “Book”, what is *MAX(global gross)*?
Q3		Draw a line chart that shows *AVG(production cost)* over *age*.
Q4	Hard	Except all movies whose *origin* is “Comic Books”, which *creative type* has the maximum *MIN(domestic gross)*?
Q5		Except all movies whose *genre* is “Adventure” or *age* is “20s”, what is *AVG(Rotten Tomatoes score)* of movies whose *MPAA rating* is “G”?
Q6		Draw a line chart that shows *AVG(IMDB score)* over *age* and separate the lines by *MPAA rating* using color.

Table 3. Six questions on a movie dataset in Task 1. We derived 6 common questions with two difficulties. Note that Q3 and Q6 required a specific chart (i.e., a line chart) instead of a single value.

contrast, Q3 and Q6 required to create a line chart to cover the cases where people change the default visual encoding (i.e., bar charts) to a different one.

For Task 2, we asked participants to freely explore the given data for 10 minutes and capture meaningful charts they found during the exploration. This approach is similar to the evaluation of Voyager system [35], where study participants explored data and tried to find as many charts as possible that are worth sharing with colleagues. Since our study participants were students, we asked them to create charts worth being included in a report for course assignments. To check correctness of the created charts, we also asked them to briefly explain the charts after Task 2.

We prepared three different datasets: a Korean SAT dataset for tutorial videos and six practice questions, an American movie dataset for Task 1 (taken from the code repository of a previous study [35]), and an academic performance dataset from a Portuguese school for Task 2 [29]. To ensure compatibility between datasets, we trimmed all datasets to have 200 rows and 12 columns. The Korean SAT dataset contained scores and grades of 200 students (i.e., region, gender, the scores and grades for five subjects). The movie dataset consisted of classification of movies (e.g., genre, creative type, and MPAA rating), budget, total gross, and review ratings. The academic performance dataset contained the performance of 200 students with categorical survey results (e.g., gender, parents' education levels, and weekly study time), age, and scores (i.e., midterm, final, and total).

Procedure

After signing a consent form and answering a screening questionnaire, participants were briefly introduced to the overall procedure and tasks for about five minutes. Then, they had a training session where they first watched three videos played on Presenter; one introductory video about pivot transformations and aggregate functions, and two tutorial videos that briefly explain the given interface. We controlled the contents and length of the videos and all videos took about three minutes. After watching each of the last two tutorial videos, participants practiced what they learned from the video on Interface following seven or eight step-by-step instructions.

After participants watched the videos, we explained how to use the session manager app (i.e., the web app on Presenter) for Task 1. Presenter showed each question on one by one. After participants read and understood the question, they tapped on the start button below the question, and then they were allowed to use the given interface to find an answer. Once they found the answer, they tapped on the corresponding choice among five choices and the done button to finish the trial on Presenter. We measured the time between 1) when they pressed the start button and 2) when they tapped on the last choice right before tapping on the done button, and considered the interval as task completion time. For questions that accept a chart as an answer (i.e., Q3 and Q6), participants only tapped on the done button without selecting a choice. The session manager app recorded task completion time and answers. Participants were allowed to familiarize themselves with the session manager app and the given interface using six exercise questions on a practice dataset. To motivate participants, we rewarded the one who completed Task 1 the fastest and the most accurate with an extra US$10. After participants finished Task 1, we gave them a two-minute break before Task 2.

In Task 2, we asked participants to explore the data to find the charts that they thought meaningful and capture them as many as possible for 10 minutes. We warned the participants that incorrect charts (i.e., charts that do not support their findings) would not be counted so that they should capture charts with care. Similarly, we also informed them that too similar charts (e.g., charts that have a similar trend) would be counted once. After the exploration, they reviewed their charts one by one with the experimenter. The participants explained 1) why they created each chart and 2) what they found from the chart. The experimenter transcribed participants' explanation. Similar to Task 1, we rewarded the one who created the most charts with extra US$10.

Upon their completion of Task 1 and Task 2, participants responded to an SUS (System Usability Scale) questionnaire [2], and we gathered subjective feedback. The entire session took about an hour per participant.

Results

We report the results of our study in three parts: (1) comparing accuracy and task completion time of Task 1, (2) counting and categorizing charts that participants created during Task 2, and (3) analyzing qualitative feedback.

Accuracy (Task 1). We first examined how each interface affected answer accuracy in Task 1; Figure 4 shows the accuracy according to the interface and difficulty conditions. We conducted a Kruskal-Wallis H test to compare the answer accuracy of the three interfaces for the two levels of difficulty. We did not find a significant difference between the interfaces in terms of answer accuracy, regardless of difficulty ($\chi^2(2) = 5.68$, *ns* for the easy condition and $\chi^2(2) = 3.32$, *ns* for the hard condition).

Completion Time (Task 1). We were especially interested in whether TouchPivot would be faster than the other two interfaces. First, we excluded the task completion time of incorrectly answered trials from analysis. For trials where participants could not choose an answer within a timeout limit, we regarded the task completion time of such trials as the corresponding timeout limit (i.e., 60 seconds for easy questions and 90 seconds for hard questions).

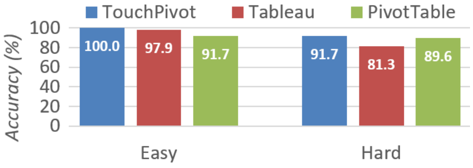


Figure 4. Answer accuracy by difficulty and interface. We did not find a significant difference between interfaces.

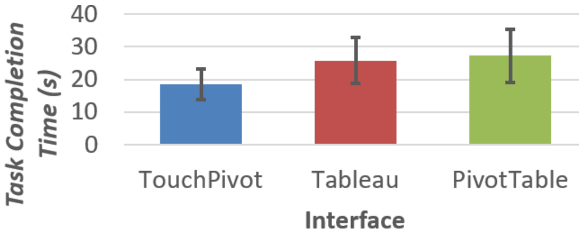


Figure 5. Mean task completion time by interface. Participants with TouchPivot answered the questions significantly faster than those with other interfaces. Error bars indicate ± 1 standard deviation.

Then, we applied a log transformation to the task completion time since the distribution of time was skewed to the right. We confirmed the distribution followed a normal distribution for each difficulty and interface condition after the log transformation (through the Shapiro-Wilk test, *ns*). Then, we applied a robust outlier detection technique [19] to identify outlying trials. We used moderately conservative criteria from a previous study [21] (i.e., measurements that differ by 2.5 * *MAD*, median absolute deviation, or more from the median were rejected). Using this procedure, we discarded task completion time of three participants (two participants from PivotTable and one from Tableau), due to their longer completion time.

We compared the task completion time of three interfaces using a mixed ANOVA. We set the difficulty factor as a within-subjects factor and the interface factor as a between-subjects factor. The mixed ANOVA revealed a significant difference in task completion time between the interfaces (*F*(2, 42) = 9.930, *p* < .001, Figure 5). Post-hoc tests using the Bonferroni correction revealed that TouchPivot (*M* = 18.41s, *SD* = 4.75s) was significantly faster than Tableau (*M* = 25.69s, *SD* = 7.06s) (*p* = .002) and PivotTable (*M* = 27.25s, *SD* = 8.07s) (*p* < .001). We did not find a significant difference between Tableau and PivotTable.

As expected, participants took significantly longer time to answer hard questions (*M* = 31.16s, *SD* = 11.61s) than easy questions (*M* = 16.4s, *SD* = 5.52s) (*F*(1, 42) = 196.31, *p* < .001). On average, participants took the longest time to answer Q5 (*M* = 38.83s, *SD* = 20.69s) which required filtering on two columns. There was no significant interaction effect between interface and difficulty factors (*F*(2, 42) = .83, *ns*).

Chart Count (Task 2). We regarded the number of charts created in Task 2 as a proxy for the amount of knowledge [35]. Table 4 shows the number of charts participants created with each interface.

As mentioned above, we did not count incorrect charts. We classified these excluded (incorrect) charts into three categories: *oversight aggregation, invalid pivot*, and *count misinterpretation*. First, the oversight aggregation category included charts whose authors (i.e., the participants who captured the charts) did not change the default aggregate function (i.e., sum) to an appropriate one (e.g., average). For example, if a chart showed the sum of scores but the author stated that the mean was intended, we classified this error as oversight aggregation. Next, the invalid pivot category included charts that pivot raw data by a numerical column instead of a categorical one. Charts in this category tended to have a huge number of items on the horizontal axis. Finally, the count misinterpretation category included charts that the participants misinterpreted; the meaning of a chart was different from the participants' understanding. Interestingly, this type of errors happened only in PivotTable and all resulted from the misunderstanding of the count aggregate function. Table 4 shows the distribution of the discarded charts over the three exclusion criteria.

Cleaning Criteria	TouchPivot	Tableau	PivotTable
Raw count	**241**	**230**	**205**
Remove duplicates	2 (239)	6 (224)	0 (205)
Oversight aggregation	15 (224)	23 (201)	29 (176)
Invalid pivot	0 (224)	2 (199)	15 (161)
Count misinterpretation	0 (224)	0 (199)	36 (125)
Count similar charts once	30 (194)	28 (171)	13 (112)
Exclude charts from those who created them blindly	0 (194)	47 (124)	27 (85)
# of charts per participant	12.13	8.27	5.67

Table 4. The number of charts participants created with each interface. After removing duplicates, we cleaned the results following three criteria: oversight aggregation, invalid pivot, and count misinterpretation. Numbers in parentheses are the number of remaining charts after each cleaning criterion.

As we informed to participants, we counted only once for the charts that are highly similar to each other. We excluded 30 charts from TouchPivot, 28 from Tableau, and 13 from PivotTable. We also excluded charts from two participants who were observed to capture many charts blindly to receive an extra reward (P41 captured 47 charts using Tableau and P19 captured 27 charts using PivotTable for 10 minutes). Note that the cleaning process was deterministic because all the cleaning criteria were rule-based. To minimize human errors, two researchers conducted the entire process together.

After the cleaning process, we compared the number of meaningful charts by interface. Through a Kruskal-Wallis H test, we found a significant difference in the number of charts between interfaces ($\chi^2(2)$ = 11.34, *p* = .003). A post-hoc analysis, Dunn's nonparametric comparison, revealed that participants with TouchPivot found more charts than those who used PivotTable (*p* = .002). We did not find a significant difference between the other two combinations.

Content of Charts (Task 2). We categorized all created charts according to their types (e.g., bar, line) and contents (i.e., the names of pivoted columns and aggregated columns). Table 5 shows the five most frequently created charts. The most frequent chart type was a bar chart, with which 20 out of 46 participants (note that we dropped results from two participants) checked how *weekly study time* affected *total*

Chart Summary	Total	Touch-Pivot	Tableau	Pivot-Table
AVG(total score) by *weekly study time*	20 (15)	10 (6)	5 (5)	5 (4)
AVG(total score) by *mother's education level*	16 (9)	9 (6)	3 (3)	4 (0)
AVG(total score) by *father's education level*	15 (8)	7 (5)	4 (3)	4 (0)
AVG(absences) by *weekly alcohol consumption*	11 (9)	3 (3)	4 (3)	4 (3)
AVG(absences) by *health*	11 (7)	5 (4)	6 (3)	0

Table 5. Top five most created charts in Task 2. Numbers in parentheses represent the number of bar charts in each category; the rest of the charts were all line charts.

score. More than a half of created charts were bar charts (271 / 403, 67.2%), and line charts followed (111 / 403, 27.5%).

Usability Scale. We compared the overall SUS score of each interface through a one-way ANOVA. We did not find a significant difference in total SUS scores between the interfaces (M = 78.28 for TouchPivot, M = 79.29 for Tableau, and M = 69.38 for PivotTable, $F(2, 47) = 2.97$, *ns*). A one-way ANOVA on each SUS question revealed significant differences in the second and sixth questions (SUS2: I found the system unnecessarily complex and SUS6: I thought there was too much inconsistency in this system). Post hoc tests using the Bonferroni correction revealed that PivotTable was unnecessarily more complex than TouchPivot ($p = .036$) and had too much inconsistency than Tableau ($p < .001$).

Freeform Feedback. Among 16 participants who used TouchPivot, 11 participants expressed positive or neutral feedback about bimanual interaction. However, eight participants expressed concerns (e.g., the overshooting problem) on using the fan menu with a non-dominant hand. For example, P9 stated "*Although left-hand interactions [using the fan menu] helped me explore the data, I didn't use it for Task 1. It was because I was asked to answer as fast as I could. I wasn't sure if I could use those interactions fast without mistakes,*" who mostly tapped on column headers to select a column. In addition, four participants preferred the WIMP-style filter view over the strikethrough interaction because of familiarity and accessibility.

During training session, TouchPivot provided an animated transition on the data table according to pivot transformations to help participants learn and follow the transformations. According to interviews, 11 out of 16 participants gave positive responses about the animation. For example, P24 expressed "*The animation may be necessary when we first learn pivot transformations. It helped me understand how rows gathered.*" However, they also stated the animation was not necessary after they became accustomed to TouchPivot. For example, P9 said "*This [animation] was great when I first learned the interface. However, if I fully learned the interface, this would delay my analysis.*" The remaining five participants were neutral. For example, P18 said *"I couldn't notice the animation.*"

DISCUSSION

Hurdles for Novices

In this section, we discuss the hurdles novices encountered during the study.

Incorrect Configuration. Pivot transformations require configuring multiple settings such as columns to pivot or aggregate and an aggregate function. Therefore, people need to remember to apply those settings and check all the settings have been set correctly. However, such a cognitive burden seemed to overwhelm novices. For example, as shown in Table 4, all three interfaces caused errors of oversight aggregation where novices forgot to configure a proper aggregate function. The most frequent cases were that novices did not change the default aggregate function (i.e., sum) although they wanted to calculate the mean. In Task 2, 29 charts (14.1%) from PivotTable and 23 charts (10.2%) from Tableau used an unintended aggregate function.

However, these errors happened less in TouchPivot (15 charts, 6.2%). From interviews, we identified two possible reasons. First, Tableau and PivotTable did not provide any explicit explanation (or summary) about what a chart was showing, while TouchPivot provided an explanatory title above the chart, such as "the mean of age by country." P4 said "*I wanted to see the mean instead of the sum, but I couldn't notice that I forgot that configuration in the [PivotTable] interface because indicators were too subtle.*" Second, Tableau and PivotTable did not persist settings applied to each column and novices often forgot this behavior. For example, people had to change the aggregate function of a column again when they removed the column from a shelf and added it again. P25 said "*It is tedious [in PivotTable] to change the aggregate function [of a column] every time I try to use the column.*"

Choosing the Right Roles. In shelf configuration interfaces, it is important to assign each column to the right role (by dragging-and-dropping a column to a shelf). However, it seems that novices have difficulty in understanding and distinguishing those roles. One frequent case was that they tried to pivot data with numbers (i.e., invalid pivot) in PivotTable, which happened 15 times. Since PivotTable initially shows all columns in a single list without any clue about types, novices had difficulty in configuring a valid pivot transformation. P11, who used PivotTable, stated "*I was confused which column [among gender and age] should be placed in which shelf [among Rows and Values].*" However, it seems that this problem could be alleviated by providing sufficient information about column types. For example, Tableau automatically conjectured column types, and showed columns explicitly in separate lists (i.e., Dimensions and Measures), which might reduce the number of such errors to two cases.

This hurdle was also observed when novices tried to apply color-encoding. For example, in Task 1, Tableau tended to be slightly faster than PivotTable in Q1 to Q5, but this trend was reversed at Q6 where novices were asked to apply color-encoding to lines in a line chart: Tableau (M = 38.77s) was more than 10 seconds slower than PivotTable (M = 27.56s). Tableau required novices to explicitly add a target column to the *Color* shelf, while PivotTable automatically applied color-encoding. This additional step may impose an extra cognitive burden on novices. Among 16 participants who used Tableau, eight participants reported issues about applying color-encoding. For example, P23 said "*I was not sure if I used color-encoding correctly.*"

Misunderstanding the Default. Count misinterpretation was the most frequent type of errors novices made during using PivotTable where they misinterpreted the meaning of a count aggregate function. For example, suppose a scenario where one is moving a categorical column, *study time*, which has five discrete levels, to the *Values* shelf in PivotTable. According to interviews, novices often expected such an interaction would calculate the average study time. However, in fact, PivotTable applies a count aggregate function to the column by default and calculates the number of students. It seems that novices were often unaware of this behavior and thus made wrong interpretation. TouchPivot, in contrast, did not allow people to apply an aggregate function to categorical columns and thus was not subject to such errors.

Lowering the Hurdles for Novices

In this section, we present possible remedies for the hurdles that we identified in the previous section.

Facilitate Correct Interpretation. We observed novices failed to create charts they intended because they were often unaware of the state of pivot interfaces or the meaning of visualizations. This result was consistent with a previous study [9] where the authors named it as *interpretation barrier*. We suggest to represent the meaning of a chart in a more natural language, not to force people to derive the meaning from the interface. In TouchPivot, we provided a more natural title, and participants with TouchPivot were more likely to create charts they intended (i.e., making less errors of oversight aggregation and count misinterpretation). For example, for a bar chart that shows the frequency of values in a categorical column, *country*, TouchPivot explicitly titled the chart 'Frequency of country.' However, in PivotTable, participants were required to derive the meaning from the interface, interpreting 'COUNT(country)' in the *Values* shelf and 'country' in the *Rows* shelf.

Consider Column Metadata. It is challenging for novices to create right data transformations and visual mappings. For example, they sometimes pivot data with numbers (i.e., invalid pivot). Therefore, it is recommended to assist them to stay on 'the right path.' For example, Tableau explicitly showed column types, and thus could reduce invalid pivot errors. As a more preventive approach, we can keep people from performing invalid data transformations and visual mappings by recommending only valid ones as in TouchPivot (e.g., no support for pivoting data with decimal numbers). It reduces the flexibility in data exploration, but people would be less overwhelmed by the interface.

Provide Tight Coupling between Data and Visualizations. TouchPivot provided tight coupling between data tables and visualizations through linking and animation. Most participants responded positively to the animation, expressing that the animation helped them understand pivot transformations. We emphasize the importance of an explicit link between data and visualizations to assist people to better understand them. Note that this may not be effective if the link is weak. For example, PivotTable also juxtaposes a pivot table with a chart but participants made many mistakes because the table and chart were not tightly connected.

Limitations and Future Work

Initially, we considered interactions that use pen and touch simultaneously, such as drawing a stroke with a pen using the right hand while holding a left finger on the screen. However, since our target device (Microsoft Surface Pro 3) did not permit the concurrent use of pen and touch, we could not fully exploit such interaction. We still believe that, if designed well, simultaneous pen and touch interactions may bring a new opportunity in data exploration on tablet devices. However, we should be aware of the complexity that such an interaction could bring in. During our study, there were participants who felt uncomfortable with bimanual interaction, even if most of our interactions did not required both hands simultaneously.

As pointed out in previous studies [1, 5], discoverability is one of the common concerns in pen and touch interfaces. To alleviate the discoverability issue, we provided visual cues where possible (e.g., semi-transparent background for the fan menu and small carets in the chart title). However, it would be interesting to see how novices discover the interactions in the three interfaces without training.

CONCLUSION

In this paper, we reveled in the emerging opportunity of pen and touch interactions to support data exploration on tablet devices. Designing TouchPivot informed by two pilot studies, we considered the hurdles of novices as the primary concern and bring in WIMP widgets to our design. We compared TouchPivot with two traditional WIMP interfaces, PivotTable from Excel and Tableau. With a short training, people could not only answer data-driven questions faster, but also create a larger number of meaningful charts with TouchPivot than others. Our results suggest blending WIMP and post-WIMP interfaces with pen and touch interaction can provide new opportunities in designing easy and less demanding interfaces on tablet devices for InfoVis novices.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2014R1A2A2A03006998). Jinwook Seo is the corresponding author.

REFERENCES

1. Dominikus Baur, Bongshin Lee, and Sheelagh Carpendale. 2012. TouchWave: kinetic multi-touch manipulation for hierarchical stacked graphs. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces (ITS '12)*, 255-264.
2. John Brooke. 1996. SUS-a quick and dirty usability scale. *Usability Evaluation in Industry* 189, 104: 4-7.
3. Jeffrey Browne, Bongshin Lee, Sheelagh Carpendale, Nathalie Riche, and Timothy Sherwood. 2011. Data analysis on interactive whiteboards through sketch-based interaction. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*, 154-157.
4. Andrew Crotty, Alex Galakatos, Emanuel Zgraggen, Carsten Binnig, and Tim Kraska. 2015. Vizdom: interactive analytics through pen and touch. In *Proceedings of the Very Large Data Base Endowment* 8, 12: 2024-2027.
5. Steven M. Drucker, Danyel Fisher, Ramik Sadana, and Jessica Herron. 2013. TouchViz: a case study comparing two interfaces for data analytics on tablets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*, 2301-2310.
6. Micheline Elias and Anastasia Bezerianos. 2011. Exploration views: understanding dashboard creation and customization for visualization novices. In *Proceedings of 13th IFIP Conference on Human-Computer Interaction (INTERACT '11)*, 274-291.
7. Mathias Frisch, Jens Heydekorn, and Raimund Dachselt. 2009. Investigating multi-touch and pen gestures for diagram editing on interactive surfaces. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*, 149-156.
8. Lars Grammel, Chris Bennett, Melanie Tory, and Margaret-Anne Storey. 2013. A survey of visualization construction user interfaces, In *EuroVis-Short Papers*, Mario Hlawitschka and Tino Weinkauf (eds.). The Eurographics Association, 19-23.
9. Lars Grammel, Melanie Tory, and Margaret-Anne Storey. 2010. How information visualization novices construct visualizations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 16, 6: 943-952.
10. Jeffrey Heer, Frank Van Ham, Sheelagh Carpendale, Chris Weaver, and Petra Isenberg. 2008. Creation and collaboration: engaging new audiences for information visualization. In *Information Visualization*, Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, Chris North (eds.). Springer Berlin Heidelberg, 92-133.
11. Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. 2010. Pen + touch = new tools. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10)*, 27-36.
12. Dandan Huang, Melanie Tory, Bon Adriel Aseniero, Lyn Bartram, Scott Bateman, Sheelagh Carpendale, Anthony Tang, and Robert Woodbury. 2015. Personal visualization and personal visual analytics. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 21, 3: 420-433.
13. Petra Isenberg, Tobias Isenberg, Tobias Hesselmann, Bongshin Lee, Ulrich Von Zadow, and Anthony Tang. 2013. Data visualization on interactive surfaces: a research agenda. *IEEE Computer Graphics and Applications (CG&A)* 33, 2: 16-24.
14. Jaemin Jo, Bongshin Lee, and Jinwook Seo. 2015. WordlePlus: expanding wordle's use through natural interaction and animation. *IEEE Computer Graphics and Applications (CG&A)* 35, 6: 20-28.
15. Bongshin Lee, Petra Isenberg, Nathalie Henry Riche, and Sheelagh Carpendale. 2012. Beyond mouse and keyboard: expanding design considerations for information visualization interactions. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 18, 12: 2689-2698.
16. Bongshin Lee, Rubaiat Habib Kazi, and Greg Smith. 2013. SketchStory: telling more engaging stories with data through freeform sketching. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 19, 12: 2416-2425.
17. Bongshin Lee, Greg Smith, Nathalie Henry Riche, Amy Karlson, and Sheelagh Carpendale. 2015. SketchInsight: natural data exploration on interactive whiteboards leveraging pen and touch interaction. In *Proceedings of 2015 IEEE Pacific Visualization Symposium (PacificVis '15)*, 199-206.
18. Sukwon Lee, Sung-Hee Kim, Ya-Hsin Hung, Heidi Lam, Youn-ah Kang, and Ji Soo Yi. 2016. How do people make sense of unfamiliar visualizations?: a grounded model of novice's information visualization sensemaking. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 22, 1: 499-508.
19. Christophe Leys, Christophe Ley, Olivier Klein, Philippe Bernard, and Laurent Licata. 2013. Detecting outliers: do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology* 49, 4: 764-766.
20. Microsoft Excel 2016, Retrieved September 21, 2016 from https://products.office.com/en-us/excel
21. Jeff Miller. 1991. Short report: reaction time analysis with outlier exclusion: bias varies with sample size.

The Quarterly Journal of Experimental Psychology 43, 4: 907-912.

22. Qlik - Data Visualization Tools, Retrieved September 21, 2016 from http://www.qlik.com/us/
23. Donghao Ren, Tobias Höllerer, and Xiaoru Yuan. 2014. iVisDesigner: expressive interactive design of information visualizations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20, 12: 2092-2101.
24. Jeffrey M. Rzeszotarski and Aniket Kittur. 2014. Kinetica: naturalistic multi-touch data visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*, 897-906.
25. Ramik Sadana and John Stasko. 2014. Designing and implementing an interactive scatterplot visualization for a tablet computer. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces (AVI '14)*, 265-272.
26. Ramik Sadana and John Stasko. 2016. Designing multiple coordinated visualizations for tablets. *Computer Graphics Forum* 35, 3: 261-270.
27. Ramik Sadana and John Stasko. 2016. Expanding selection for information visualization systems on tablet devices. In *Proceedings of the 2016 ACM Conference on Interactive Surfaces and Spaces (ISS '16)*, 149-158.
28. Sebastian Schmidt, Miguel A. Nacenta, Raimund Dachselt, and Sheelagh Carpendale. 2010. A set of multi-touch graph interaction techniques. In *Proceedings of the 2010 ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*, 113-116.
29. Student Performance Data Set. Retrieved September 21, 2016 from https://archive.ics.uci.edu/ml/datasets/Student+Performance
30. Tableau Software, Retrieved September 21, 2016 from http://www.tableau.com/
31. Vizable by Tableau. Retrieved September 21, 2016 from https://vizable.tableau.com/
32. Julie Wagner, Stéphane Huot, and Wendy Mackay. 2012. BiTouch and BiPad: designing bimanual interaction for hand-held tablets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*, 2317-2326.
33. Jagoda Walny, Bongshin Lee, Paul Johns, Nathalie Henry Riche, and Sheelagh Carpendale. 2012. Understanding pen and touch interaction for data exploration on interactive whiteboards. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 18, 12: 2779-2788.
34. Colin Ware. 2012. *Information Visualization: Perception for Design*. Elsevier.
35. Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2016. Voyager: exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 22, 12: 649-658.
36. Emanuel Zgraggen, Robert Zeleznik, and Steven M. Drucker. 2014. PanoramicData: data analysis through pen & touch. *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20, 12: 2112-2121.