Empirical Comparison of Dynamic Query Sliders and Brushing Histograms

Qing Li, Chris North
Center for Human-Computer Interaction
Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061 USA
{qili2, north}@cs.vt.edu
http://infovis.cs.vt.edu/

Abstract

Dynamic queries facilitate rapid exploration of information by real-time visual display of both query formulation and results. Dynamic query sliders are linked to the main visualization to filter data. A common alternative to dynamic queries is to link several simple visualizations, such as histograms, to the main visualization with a brushing interaction strategy. Selecting data in the histograms highlights that data in the main visualization. We compare these two approaches in an empirical experiment on DataMaps, a geographic data visualization tool. Dynamic query sliders resulted in better performance for simple range tasks, while brushing histograms was better for complex trend evaluation and attribute relation tasks. Participants preferred brushing histograms for understanding relationships between attributes and the rich information they provided.

CR Categories: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – Information Filtering, Query Formulation; H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/Methodology

Keywords: Dynamic query, slider, histogram, usability study, information visualization, multidimensional visualization

1 Introduction

In information visualization, dynamic queries allow users to rapidly formulate queries with graphical widgets, such as sliders, for direct manipulation of databases. At the introduction of dynamic queries, initial evaluations proved their effectiveness over less dynamic display methods [Ahlberg et al. 1992]. One problem with the initial design occurs when data is not evenly distributed. Small adjustments to dynamic query sliders can suddenly filter most of the data from the display, resulting in user disorientation.

Many alternative designs have been proposed, including Data Visualization Sliders [Eick 1994], Magic Lens Filter [Fishkin and Stone 1994], Attribute Explorer [Tweedie et al. 1994], Influence Explorer [Tweedie et al. 1996], VQuery [Jones 1998], Dynamic Histograms [Derthick et al. 1999], Parallel Bargrams [Wittenburg et al. 2001], and Descarts [Andrienko and Andrienko 1999]. In general, two competing strategies have emerged: dynamic query (DQ) sliders, and brushing histograms.

IEEE Symposium on Information Visualization 2003,
October 19-21, 2003, Seattle, Washington, USA
0-7803-8154-8/03/$17.00 ©2003 IEEE

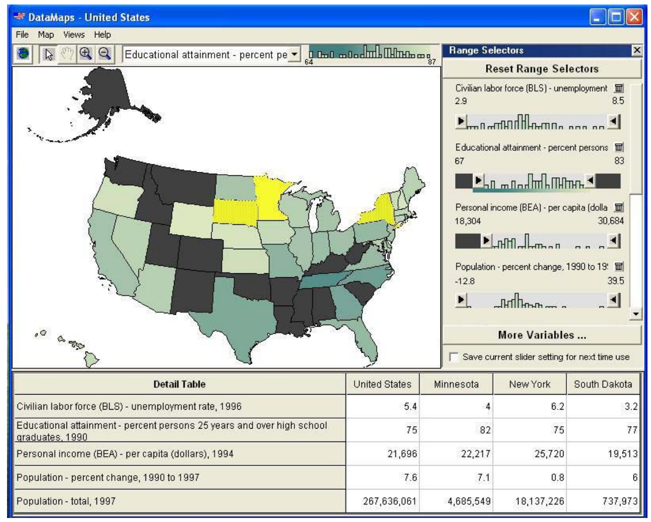


Figure 1: DataMaps user interface with DQ sliders

We implement these two strategies into DataMaps, a generalized geographic information visualization tool [Dang et al. 2001] (Figure 1). We then compare these two strategies in an empirical experiment to determine their strengths and weaknesses.

The work presented in this article consists of five parts. First, we review the related work and summarize lessons we have learned. In the second part, the prototype design for DQ slider and brushing histogram is illustrated. Their merits and limitations are discussed in detail. Third, we describe our empirical study to compare these two query tools. A within subjects, counterbalanced design is used in our test. The collected data is visualized, which helps us identify the general trends of data, and then statistical analysis is applied for rigorous evaluation. Finally, we make conclusion and direct future work to improve the efficiency of brushing histograms.

2 Literature Review

Dynamic queries enable interactive visualization of multidimensional data to facilitate decision making. Significant amount of research has explored appropriate combination of visual representation for query widgets and result display. Ahlberg and Shneiderman [1994] use sliders as the query tool in the Homefinder and the Filmfinder (Figure 2). The sliders provide visible limits on the query ranges and graphical representation of the database and the query results. They are tightly coupled to filter primary visualizations, and support rapid and incremental actions. Ideas proposed by Eick [1994] include providing an interactive color scale within query sliders, using a barplot for discrete data and a density plot for continuous data.

Proceedings of the IEEE Symposium on Information Visualization 2003 (INFOVIS'03)
0-7695-2055-3/03 $ 17.00 © 2003 IEEE

Figure 2: Query Slider

Tweedie et al. [1996] introduce a graphical interactive tool named Influence Explorer, which can be used for examining relationships within multi-attribute datasets. This tool employs interactive histograms to represent each attribute. This one-dimensional view simplifies the understanding of relationships within the dataset. Influence Explorer supports qualitative and quantitative exploration. Users can set query criteria by defining attribute ranges with sliders. This action leads to the "color linking" of those items that lie within the selected range on all related histograms. It also allows users to define exploratory limits on parameters to invoke "color coding", e.g. in Figure 3, red color coding identifies items that lie within all the performance limits, black ones are associated with one limit failure. Relaxation of the upper limit will cause some black items to change to red color, thus providing an intuitive way to make decisions on a particular upper limit.

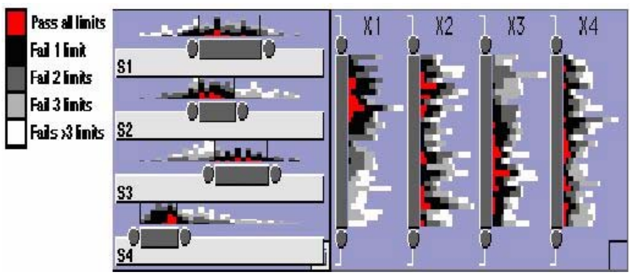


Figure 3: Influence Explorer - setting up limits on the performance histograms [Tweedie et al. 1996]

Parallel bargrams are chosen to visualize data for Internet shopping in [Wittenburg et al. 2001]. Each bargram is associated with an information dimension from the underlying model. Users can interactively select value ranges along the bargrams in order to reveal hidden relationships as well as query and restrict the set through direct manipulation. A focus + context view is afforded in which detail about individual items is revealed within the context of the global multidimensional attribute space by brushing and marking interaction (Figure 4).

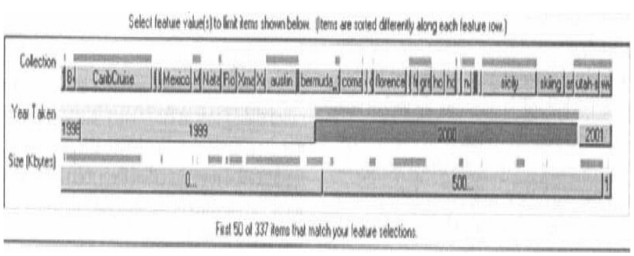


Figure 4: Parallel bargrams are used to visualize a dataset consisting of 707 photos [Wittenburg et al. 2001]

The Interactive Maps developed by Andrienko and Andrienko [1999] concentrates on dynamic manipulation technique to support visual exploration of spatially referenced data. The software so called Descarts contains several slider units, which represent the value ranges of numeric attributes in the dataset under analysis. Each unit consists of four parts, (1) a slider line containing the slider, (2) a dot plot showing the value distribution of the presented attribute, (3) a box-and-whiskers plot offering a generalized presentation of the value distribution, and (4) a color band divided into two or more segments representing classified data. The system provides brushing and linking functions to support Dynamic Comparison, which favors revealing spatial patterns, and Dynamic Classification, which is helpful for grouping and observing spatial distributions of groups of objects (Figure 5).

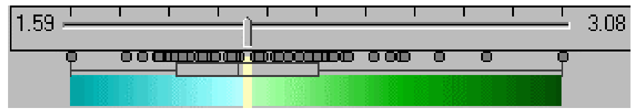


Figure 5: A slider in Descarts [Andrienko and Andrienko 1999]

[SAS JMP] is a commercial data analysis tool for interactive statistical graphics, implementing dynamic linkage between data tables and graphs. One of the data displays is the distribution histogram. It supports multiple-bar selections, granularity control and zoom level adjustment. The selected data will be highlighted in the data table and vice versa. Enhancing statistical analysis with information visualization, JMP is widely used in business and research fields (Figure 6).

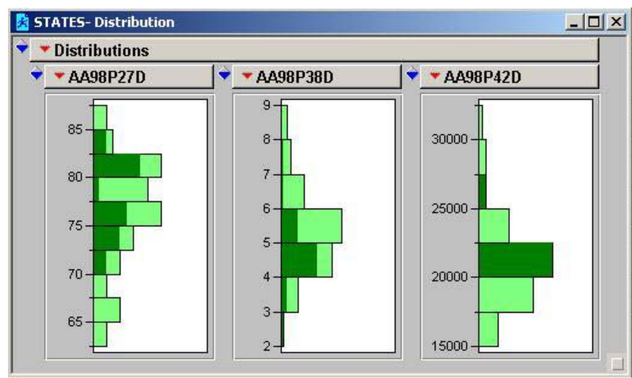


Figure 6: JMP Distribution Histograms

In summary, three lessons have been learned from previous works:

- Both query sliders and brushing histograms can improve the directness of the interaction.
- Histogram, bargram and dot plot are simple and powerful visual representation to reveal relationships among multiple attributes within a dataset, when linked for brushing.
- There is a trade-off between the richness of information, simplicity and accuracy.

For both research and commercial data analysis tools, little work has been done for usability evaluation or comparison. It will be interesting to tailor dynamic query tools to tightly incorporate them into applications used in everyday life and evaluate them in a user-centered design. However, further guidance is needed about which DQ tools are more effective for different tasks.

3 Prototype Design

DataMaps is a geographic information visualization tool that will be widely distributed on the Census Bureau's forthcoming "Counties USA" CD-ROM for use in government agencies, schools and businesses. It links the data from the U.S. Census Bureau with geographic locations (U.S. states and counties). With the help of this dynamic query tool, users can perform tasks such as comparing states with multiple criteria, finding interesting patterns of the dataset and tradeoffs. Auxiliary functions include pan and zoom the map, a spreadsheet showing detail table at the bottom, and a color-the-map function, which allows users to color regions in the map according to their values for a given attribute.

3.1 Dynamic Query Sliders

The DataMaps prototype with DQ sliders is shown in Figure 1. The data consists of several data attributes representing census statistics for each of the 50 U.S. states. Each DQ slider is a double-box slider widget representing one of the data attributes, and filters states based on user-specified minimum and maximum attribute values. The criteria of all sliders are conjunct. Filtered states are colored dark gray in the map. Users can select states from the map to view the detail table at the bottom. Each DQ slider is augmented with a static histogram showing data distribution for the attribute (Figure 7). Moving the mouse over a bar in a histogram displays a tooltip indicating the attribute value range for that bar, and the number of states within that range. The labels above sliders indicate the current data range. For example, the resulting query in Figure 7 is those states whose educational attainment is more than 70% and unemployment rate is less than 7%.

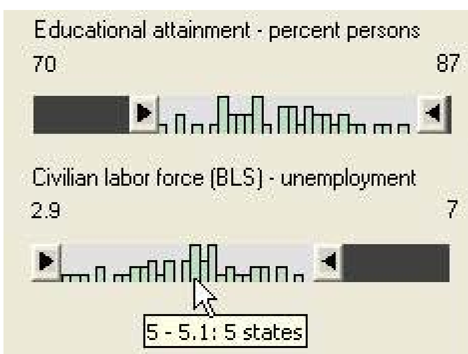


Figure 7: DQ Sliders

DQ slider is considered to be a very efficient tool for dynamic queries. There is less need for users to acquire specific knowledge about the structure of the database before making a query. The feedback is quick and natural to users (i.e. the results will be immediately shown when scrolling sliders). However, when the data is not evenly distributed, for example, when 99% data concentrates on the lowest side, a small change on the slider can suddenly filter almost all states from the display, resulting in user disorientation. We add a histogram on each slider to mitigate this effect. Users will have the overall image for data distribution before making queries.

3.2 Brushing Histograms

An alternate version of DataMaps replaces the DQ sliders with brushing histograms, which is similar to the JMP distribution histograms. There is interactive tight coupling (Select $\Leftrightarrow$ Select) between the map and histograms and among histograms [North 2001]. Users can directly select bars in the histogram to highlight the corresponding states in the map, and vice versa. The bidirectional coupling helps people to evaluate selected regions by displaying their distribution in histograms. Corresponding portions of bars are also highlighted in the other histograms. For example, in Figure 8, selecting the high range of educational attainment shows the higher portion of income. Tooltips indicating the sub-range and the number of states within that range will also be shown as users move the cursor over histogram bars. A Reset button is provided on top of the panel for resetting all histograms. The height of bars is proportional to the number of states within each sub-range, however, when a bar is too small to be seen, we assign a minimum height value to make it visible to users. The height of histograms is slightly larger than DQ sliders to facilitate direct selection of histogram bars.

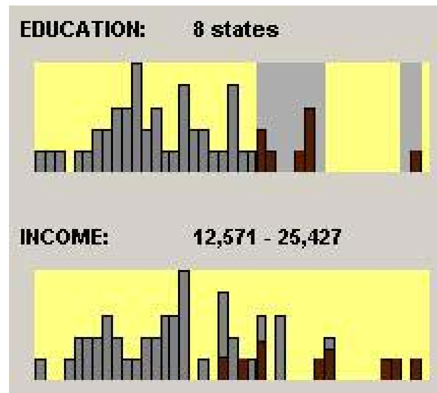


Figure 8: Brushing Histograms

Conceptually, brushing histogram is an opposite approach from DQ sliders. Four major differences are identified:

- *Filtering vs. highlighting.* Brushing histogram users highlight data of interest; DQ slider users filter undesired data.
- *Single range query vs. multiple ranges query.* DQ sliders allowed users to select only a single range, while brushing histograms allowed users to select multiple discontinuous ranges for a given attribute.
- *Interacting with query vs. interacting with data.* With DQ sliders, users interact with the query by specifying lower and upper bounds of the data range using the slider thumbs. In contrast, by manipulating histogram bars, brushing histogram users interact directly with data.
- *One directional interaction vs. bi-directional interaction.* DQ slider visualizes the query formulation (input) while brushing histogram displays the query results, too (input and output).

4 Usability Experiment

The study consisted of two independent variables: (1) the type of query tool (DQ sliders vs. brushing histograms) and (2) the type of tasks. The dataset contained 6 attributes, including educational attainment, personal income, number of crimes, population, median rent and number of farms, and 50 records for 50 US states.

Six tasks were designed, including single range, multiple ranges, multiple criteria, attribute correlation, compare, and trend evaluation tasks. All tasks were in the form of multiple-choice questions (Table 1).

Task Name	Description
Single range	Finding states within a single range for a given attribute. Example: How many states have the population between 20 and 25 millions in 1996?
Multiple ranges	Finding states within multiple ranges for a given attribute. Example: List the number of states with population in the following ranges: 6.3 – 10 millions, 6.3 – 14 millions and 6.5 – 18 millions.
Multiple criteria	Finding states according to different ranges on multiple attributes. Example: How many states have the number of farms within 28,000 – 85,000 and the population more than 10 millions?
Attribute correlation	Discovering the correlation between two attributes. Example: What's the relationship between educational attainment and personal income? Potential answers include: no relationship, direct proportion or inverse proportion.
Compare	Comparing states according to multiple criteria. Example: Given three states, which one has the lowest median rent?
Evaluate trend	Evaluating the trend of a particular state in the global context. Example: What kind of state is Florida in the United States? The potential answer could be that Florida had relatively higher population and median level of income compared with other states.

Table 1: Task Scenarios

The dependent variables included user performance time to complete each task, correctness of their answers, and user satisfaction ratings. 36 technical undergraduate and graduate students participated in the test. 20 of them performed five tasks without multiple criteria task and the other 16 users completed all six tasks. A within subjects, counterbalanced design was used. At the end of the test, participants completed a post-test questionnaire to rate their satisfaction on the two query tools based on the performed tasks. In preparation for this experiment, we conducted a pilot study with four participants to discover bugs in prototypes, and refined the tasks and questionnaires.

All test sessions were conducted in a usability study lab. Each participant used a Windows XP workstation on which the DataMaps prototypes with DQ sliders and brushing histograms run. The CPU speed of computers is 1.70 GHz. Test materials were generated for communication with the participants on testing requirements and data collection. Materials used for testing included:

- Orientation
- Two sets of similar tasks for different query tools
- Data collection form
- Post-test questionnaires for each query tool

4.1 Testing Procedure

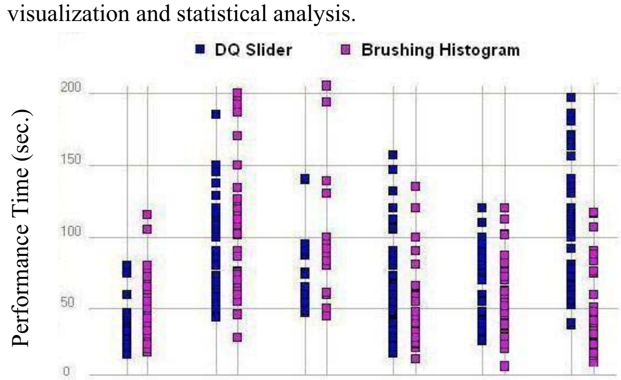
First, participants were given an introduction about the purpose and nature of the testing by the test administrator. Next, the participants were given two sample tasks for exercise. The administrator gave and explained the answers when completed.

Third, each participant was involved in a single test session that lasted for about thirty minutes. They were given written task scenarios on paper and asked to perform the tasks. Half of them used DQ slider first and then brushing histogram and the other half used them in the inverse order. Participants were encouraged to "think aloud" as they performed tasks. The test administrator allowed participants to take time to figure out how to accomplish the tasks. If they became lost or confused, the administrator asked questions in an attempt to discern the underlying causes of the difficulty. The time limit for each task was four minutes. If the participant failed to find a method to solve a task within one minute, hints about how to complete the task were given so that their ability to do the task could be assessed. Data Loggers recorded the time and the strategies that subjects took to finish each task, and also took notes on users' feedback. In particular, we paid attention to their comments on which features were easy or difficult, which features were clear or confusing and why, and which features were important or not.

At the end of the test, the participants were asked to complete a post-test questionnaire to rate their satisfaction on the two types of query tools based on the performed tasks.

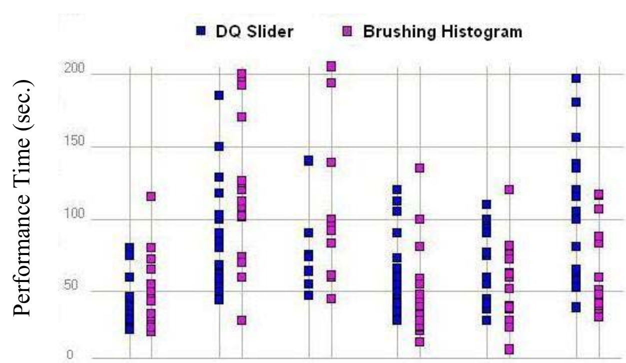
4.2 Data Analysis

Two methods were used in our data analysis: information visualization and statistical analysis.



Task Name | Single Range | Multiple Ranges | Multiple Criteria | Attribute Correlation | Compare | Evaluate Trend

Figure 9: Visualization of performance time (measured in seconds) for all subjects on DQ sliders and brushing histograms.



Task Name | Single Range | Multiple Ranges | Multiple Criteria | Attribute Correlation | Compare | Evaluate Trend

Figure 10: Visualization of only the first round for DQ sliders and brushing histograms.

First, data was visualized with SpotFire [Ahlberg 1996]. The performance time of six tasks using DQ sliders and brushing histograms was shown in Figure 9. We observed that DQ slider users generally took less time to perform single/multiple range(s) and multiple criteria tasks. Brushing histograms were more efficient at performing attribute relation, compare and trend evaluation tasks. To avoid the learning affect caused by the within-subject test, we further picked the data only for users' initial performance. The result was shown in Figure 10, which conformed to the general trends observed in Figure 9. In the first 3 tasks, there were several outliers with slow performance, most of which used brushing histograms first. This might indicate that using DQ sliders first acted as a useful learning step towards using brushing histograms.

Second, statistical analysis method was employed. Figure 11 showed the mean user performance time and answer correctness for all 12 treatment combinations. First, a two-way ANOVA analysis on performance time indicated a main effect of query tool and task type, and an interaction effect between query tool and task, all significant at $p<0.05$. For correctness, there was a main effect of task type and an interaction effect between query tool and task for correctness at $p<0.05$ (Table 2).

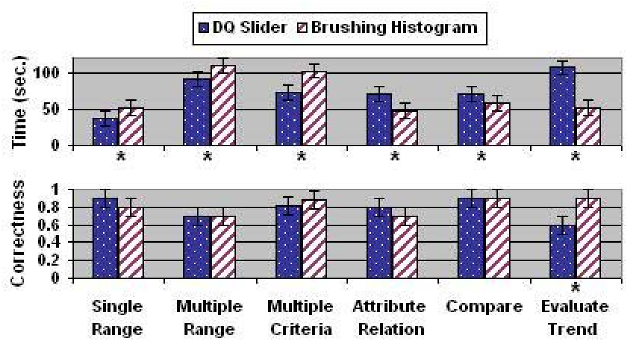


Figure 11: Mean user performance time and correctness for each task and query tool. Asterisks indicate significant difference at $p<0.05$. Correctness: 1 = right, 0 = wrong

Source of variation	SSq	DF	MSq	F	p
Tool	12145.2	1	12145.2	12.38	0.0005
Task	130157.1	4	32539.3	33.17	<0.0001
Tool × Task	65157.1	4	16289.3	16.61	<0.0001
Within cells	343332.4	350	980.9		
Total	550791.9	359			

a. PerformanceTime

Source of variation	SSq	DF	MSq	F	p
Tool	0.0	1	0.0	0.02	0.8952
Task	2.6	4	0.6	4.03	0.0033
Tool × Task	2.0	4	0.5	3.19	0.0136
Within cells	55.9	350	0.2		
Total	60.5	359			

b. Correctness

Table 2: Two-way ANOVA over the 2 × 5 independent variable matrix. Multiple criteria task is not included in order to avoid unbalanced analysis. SSq: Sum of squares, DF: Degrees of Freedom, MSq: Mean square, F: Ratio of Mean square/Mean square within, p: probability

Paired samples t-tests on query tools for each task type gave further insight (Table 3). For the single/multiple range(s) and criteria tasks, DQ sliders resulted in significantly faster performance time to complete the tasks. Brushing histograms resulted in significantly faster performance time to complete compare, attribute correlation and trend evaluation tasks ($p<0.05$).

Mean	Single Range	Multiple Range	Multiple Criteria	Attribute Relation	Compare	Evaluate Trend
DQ Slider	37.5	91.3	72.8	70.6	70.4	106.6
Brushing Histogram	51.9	109.4	102.1	47.8	57.3	51.4
p	0.0013	0.0179	0.0231	0.0014	0.0306	<0.0001

a. Time

Mean	Single Range	Multiple Range	Multiple Criteria	Attribute Relation	Compare	Evaluate Trend
DQ Slider	0.9	0.7	0.8	0.8	0.9	0.6
Brushing Histogram	0.8	0.7	0.9	0.7	0.9	0.9
p	0.0831	0.7859	0.5805	0.3731	0.5710	0.0103

b. Correctness

Table 3: T-test on performance time and correctness

Mean	1. Organization of information on the system Confusing 1 2 3 4 5 Very clear	2. Learning to operate the system Difficult 1 2 3 4 5 Easy	3. System speed: Too Slow 1 2 3 4 5 Fast Enough	4. Performing an operation leads to a predictable result: Never 1 2 3 4 5 Always	5. Help to identify relationships between attributes: Not Useful 1 2 3 4 5 Very Useful	6. Setting query data range as required: Rough 1 2 3 4 5 Exact	7. Overall reactions to the system: Frustrating 1 2 3 4 5 Satisfying
DQ Slider	3.8	4.3	4.6	3.6	3.3	3.2	3.8
Brushing Histogram	4.1	4.2	4.5	3.8	4.2	3.5	4.0
p	0.0778	0.5971	0.8004	0.8685	0.0012	0.1759	0.2178

Table 4: T-test on user satisfaction ratings

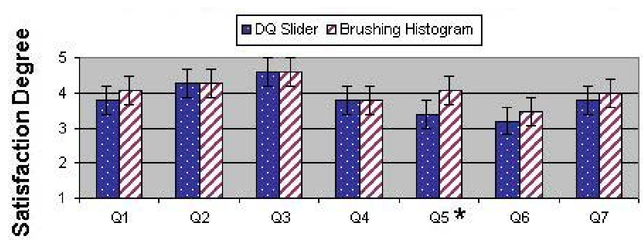


Figure 12: Mean user satisfaction ratings for each query tool. Asterisks indicate difference at $p<0.05$

Table 4 and Figure 12 showed the mean user satisfaction ratings for each query tool on 7 different survey questions. The pronounced significant difference occurred in Q5 (Helps to identify relationships among attributes?) with $p<0.05$. This result coincided well with the performance results on the more complex tasks. Users clearly recognized the additional insight provided by the brushing histograms. Q1 (Organization of information on the system?) was weakly significant in favor of brushing histograms

($p<0.1$), users thought the information provided by brushing histograms was more clear than that provided by DQ sliders.

4.3 Result Analysis

T-tests on performance time and correctness showed that DQ sliders were more efficient than brushing histograms in range and criteria tasks. We observed that three primary reasons contributed to this result. First, the smooth movement of DQ sliders had an advantage for selecting continuous ranges. With brushing histograms, users had difficulty in accurately selecting specific sub-ranges that were not directly on histogram bar boundaries. Second, the simpler interactions of DQ sliders (only the slider thumbs were interactive) helped avoid mistakes. In contrast, all bars in the brushing histograms were interactive. The targets for clicking were narrower and smaller compared with DQ sliders. Thus it was easier for users to make incorrect or accidental selections. Third, once a user selected data ranges on a brushing histogram, other histograms also reflected the selected states by highlighting the corresponding portions of histogram bars. Some users were confused by that additional feedback, especially when making queries on multiple histograms.

Brushing histograms had dominant advantages over DQ sliders in performing trend evaluation, attributes relation and compare tasks. We noticed that for trend evaluation and attribute relation tasks, the brushing histograms alone were sufficient for some users to correctly answer the questions without using the map display at all. This indicated the additional insight provided by the brushing histograms strategy, and the potential to use brushing histograms as an independent visualization component that could be embedded in various information systems. On the other hand, two-thirds of DQ slider users needed hints in order to perform such tasks. For example, in order to find the relationship between Attribute1 and Attribute2, a two-step method could be used. First, dragging the left side box of the slider of Attribute1 for a distance and identifying which states were left. Second, dragging the right side box of the slider of Attribute2 and watching if the left states were filtered out. If the answer was yes, these two attributes were direct proportional to each other. If the left states remained unchanged, this indicated a relationship of inverse proportion. Some users just moved the two sliders aimlessly or pointed to the histogram bars of each slider to compare the number of states within a certain range. In such cases, the administrator gave a hint, for instance, letting the user first identify a given state, then move each of the two sliders to see whether the state was filtered out or not. Most users were able to solve the problem after hints; however, a few users failed at last, which resulted in long performance time and incorrect answers.

An interesting phenomenon for brushing histograms was that some users attempted to perform the attributes relation task even without operating the histograms (i.e. only by observing the histograms). With regard to their comments after the test, we found the histogram bars were thought to represent states instead of sub-ranges and be ordered alphabetically along the histogram, although we explained the meaning in the orientation before each test. Some users suggested making the histogram tooltips always visible like labels. The lesson is that clear visual affordance should be provided to indicate the meaning of histograms in the user interface.

The difference between DQ sliders and brushing histograms for the compare task was also significant. We observed some users tried to use the color-the-map function provided by DataMaps when they performed the task using DQ sliders at first. But soon they found the colors for the required states were hard to differentiate. After selecting the required states, most users only referred to the detail table to make comparison. For brushing-histogram users, they chose to observe the highlighted bars in the histograms, which was faster. This indicated an advantage of the additional feedback provided by the brushing histograms as output.

Since DQ sliders and brushing histograms used opposite interaction approaches (filtering vs. highlighting), we had expected there would be some difference of satisfaction for Q2 (Learning to operate the system?) and Q4 (Performing an operation leads to a predictable result?). However, we didn't find a significant difference for these questions. We did notice in the result visualizations (Figure 9 and Figure 10) that there were more outliers (slow users) for brushing histograms than DQ sliders, especially for the first round. This implied brushing histograms were harder to learn compared with DQ sliders. No significant difference was found for Q3 (System Speed?) and Q7 (Overall reaction to the system?). For both query widgets, Q6 (Setting query data range as required?) got the lowest satisfaction ratings. Most users complained that it was difficult to query exact data ranges. Clearly, some form of fine-tuning control was needed.

5 CONCLUSION

We have presented the design of two dynamic query widgets: DQ sliders and brushing histograms, and performed a summative usability study and drawn conclusions on their merits. By using both the data visualization and statistical methods, we found that brushing histograms were superior for more complex discovery tasks (i.e. attribute correlation, compare and trend evaluation), resulting in 34% faster performance on average, and more highly rated by users for relationship identification. This indicated its ability to function on its own as an information visualization tool. On the other hand, DQ sliders were superior for more simple range specification tasks (i.e. single range, multiple ranges, multiple criteria), resulting in 24% faster performance on average, and functioned more as an auxiliary control for other visualizations.

6 FUTURE WORK

We noted that the advantage of DQ sliders (i.e. good at simple range tasks) was primarily due to fairly simple usability issues related to the specifics of the slider controls. This leads us to believe that the design of the brushing histograms strategy could be upgraded to support these capabilities, thus enabling the best of both.

In our study, only a limited dataset (the states data with 6 attributes) were chosen. It was expected that brushing histograms would meet problems when dealing with large datasets with complex data structure. We conducted an experiment in our software design stage to visualize counties data, which consisted of more than 3000 records, however, the system speed was significantly slowed down when brushing histograms were used. This was because a single histogram change would cause all of the other histograms to be redrawn. Optimizing the existing algorithm to only re-draw the matching histogram bars would be one solution, or another kind of display of distribution might be used to simplify the computational work.

During the experiment on counties data, we also found that for some attributes, most data concentrated on one side, either lowest or highest value. This resulted in only one or two column(s) visible in the histogram. This phenomenon happened in nearly 80% of attributes due to high population concentration in big cities such as Los Angeles and New York. Users might want to find out the distribution for data within one or two sub-range(s). To solve these problems, we considered adding a zooming function to histograms. Histograms were drawn in the whole range initially. Users could then set the data range for a histogram when they wanted to see the data distribution within any specific range (Figure 13). By only manipulating a subset of the data, more accurate data ranges could be acquired.

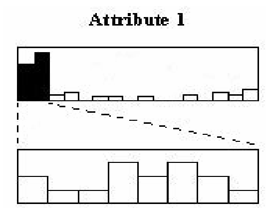


Figure 13: Zoomable histogram

In the test, we observed that users had difficulty selecting sub-ranges directly from histograms when sub-ranges were too narrow to be identified. As a result, participants often selected wrong sub-ranges. In addition, many participants complained that for both query tools, they couldn't specify accurate data ranges to query. This problem was even worse for brushing histograms since there were only a fixed number of sub-ranges. Adding granularity controls to allow users to customize the granularity of the histogram bars might be a solution to such problems. An additional solution might be to temporarily enlarge a histogram to the width of the screen when the user interacts with it.

Acknowledgements

We would like to thank the Census Bureau for supporting this research. We also appreciate Chen Song and Xiaofeng Bao for their help in the setup of the usability study.

References

AHLBERG, C. 1996, Spotfire: An Information Exploration Environment, *SIGMOD REPORT*, vol. 25 (4), 25-29

AHLBERG, C. AND SHNEIDERMAN, B. 1994, Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays*, Proc. CHI 94*, ACM Press, 313-317.

AHLBERG, C., WILLIAMSON, C., AND SHNEIDERMAN, B. 1992, Dynamic Queries for Information Exploration: An Implementation and Evaluation, *Conference proceedings on Human Factors in Computing Systems*, 1992, Monterey, California, ACM Press, 619-626

AHLBERG, C. AND WISTRAND, E. 1995, IVEE: An Environment for Automatic Creation of Dynamic Queries Applications, *CHI'95 Demonstrations*, 1995, Denver, Colorado, ACM Press, 15-16.

ANDRIENKO, G.L. AND ANDRIENKO, N.V. 1999, Interactive Maps for Visual Data Exploration, *International Journal Geographic Information Science*, vol. 13 (4), 355-374.

DANG, G. NORTH, C., AND SHNEIDERMAN B. 2001, Dynamic Queries and Brushing on Choropleth Maps, *Proc. IEEE International Conference on Information Visualization 2001*, 757-764

DERTHICK, M., HARRISON, J., MOORE A., AND ROTH, S.F. 1999, Efficient Multi-Object Dynamic Query Histograms, *Proceedings of the IEEE Symposium on Information Visualization 1999*, San Francisco, California, October, 84-91.

EICK, S.G. 1994, Data Visualization Sliders, *Proceedings of the 7th annual ACM symposium on User interface software and technology*, 1994, ACM Press, 119-120

FISHKIN, K. AND STONE, M.C. 1995, Enhanced Dynamic Queries via Movable Filters, *Conference proceedings on Human factors in computing systems*, 1995, Denver, Colorado, ACM Press, 415-420

JONES S. 1998, Graphical Query Specification and Dynamic Result Previews for a Digital Library, *Proceedings of the 11th annual ACM symposium on User interface software and technology*, 1998, San Francisco, California, ACM Press, 143-151

LI, Q., BAO, X., SONG, C. ZHANGE, J., AND NORTH, C. 2003, Dynamic Query Sliders vs. Brushing Histograms, *Extended Abstract of CHI 2003,* 2003, Fort Lauderdale, Florida, ACM Press, 834-835

NORTH, C. 2001, Multiple Views and Tight Coupling in Visualization: A Language, Taxonomy, and System, *Proc. CSREA CISST 2001 Workshop of Fundamental Issues in Visualization*, 626-632,

SAS INSTITUTE INC., JMP – Statistical Discovery Software, http://www.jmp.com.

TWEEDIE, L., SPENCE, B., WILLIAMS, D., AND BHOGAL R. 1994, The Attribute Explorer, *Proceedings of the CHI '94 conference companion on Human factors in computing systems*, 1994, Boston, Massachusetts, ACM Press, 435-436

TWEEDIE, L., SPENCE, R., DAWKES, H., AND SU, H. 1996, Externalising Abstract Mathematical Models, *Conference proceedings on Human factors in computing systems*, Vancouver, British Columbia, Canada, 1996, ACM Press, 406 - ff.

WITTENBURG, K., LANNING, T., HEINRICHS, M., AND STANTON, M. 2001, Parallel Bargrams for Consumer-based Information Exploration and Choice, *Proceedings of the 14th annual ACM symposium on User interface software and technology*, Orlando, Florida, 2001, ACM Press, 51-60