Dynamic Query Sliders vs. Brushing Histograms

Qing Li, Xiaofeng Bao, Chen Song, Jinfei Zhang, Chris North
Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061 USA
{qili2, xibao1, chsong1, jizhan17, north}@cs.vt.edu

ABSTRACT
Dynamic queries facilitate exploration of information through 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. Selecting data in the histograms highlights that data in the main visualization. We compare these two approaches in an empirical experiment on DynaMaps, a geographic data visualization tool. Dynamic query sliders resulted in better performance for simple range tasks, while brushing histograms was better for complex comparison, tradeoff, and pattern tasks. Participants preferred brushing histograms for understanding relationships between attributes.

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 [1]. 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.

Since then, many alternative designs have been proposed [2][4]. In general, two competing strategies have emerged: dynamic query sliders, and brushing histograms. We implement these two strategies into DynaMaps, a generalized geographic information visualization tool [3]. We then compare these two strategies in an empirical experiment to determine their strengths and weaknesses.

PROTOTYPE DESIGN
The DynaMaps prototype with dynamic query (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. Filtered states are colored dark gray in the map. Each DQ slider is augmented with a static histogram showing data distribution for the data attribute (Figure 2a). 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.

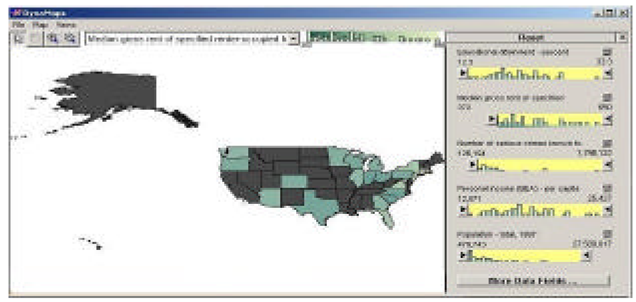

Figure 1: DynaMaps with DQ sliders

An alternate version of DynaMaps replaces the DQ sliders with brushing histograms (Figure 2b). Brushing histograms enables users to directly interact with the histogram. Users can directly select bars in the histogram to highlight the corresponding states in the map, and vice versa. Corresponding portions of bars are also highlighted in the other histograms. Conceptually, this is the opposite approach from DQ sliders. Users highlight states of interest, instead of filtering undesired states. An advantage is that users can select multiple discontinuous ranges in a histogram. In Figure 2b, three separate ranges of the first attribute have been selected, and corresponding highlights shown in the second attribute.

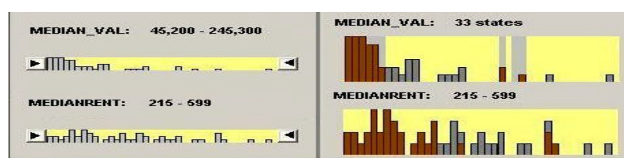

Figure 2: (a) DQ Sliders, (b) Brushing Histograms

EXPERIMENT
The study consisted of two independent variables: (1) the type of query tools (either DQ sliders or brushing histograms), and (2) the type of task. Four task types were tested: two simple counting tasks (counting states within simple attribute value ranges), find task (finding states meeting multiple tradeoff criteria), compare task (fitting a state within the distribution of all other states for multiple attributes), and pattern task (discovering patterns of data relationships between multiple attributes). All tasks were in the form of multiple-choice questions. The dependent variables included user performance time to complete

each task, correctness of their answers, and user satisfaction ratings. 20 technical undergraduate and graduate students participated. 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.

RESULTS

Figure 3 shows the mean user performance times and correctness for all 10 treatment combinations. First, a two-way ANOVA analysis on performance time indicates a main effect of query tool and task type, and an interaction effect between query tool and task, all significant at $p<0.05$. There is also an interaction effect for correctness at $p<0.05$.

T-tests on query tool for each task type gives further insight. For the range count task (Count2), DQ sliders resulted in significantly faster performance time to complete the task ($p<0.05$). Similarly, Count1 was weakly significant at $p<0.1$. We observed that three primary reasons contributed to this result. First, DQ sliders have finer granularity for selecting ranges. Hence, 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 are interactive) helped avoid mistakes. Since all bars in the brushing histograms are interactive, it was easier for users to make incorrect or accidental selections. Third, some users had difficulty identifying which states were highlighted in the brushing histograms and map. The highlight color interferes somewhat with the choropleth coloring of the map. The dark gray color of filtering with DQ sliders is much more prominent.

For the comparison task (Compare), brushing histograms resulted in significantly faster performance time to complete the task ($p<0.01$) and more correct answers ($p<0.05$). Find and Pattern tasks were weakly significant in favor of brushing histograms at $p<0.1$ for performance time. We noticed that for the pattern task, the brushing histograms alone were sufficient for some users to correctly answer the question without using the map display at all. This indicates the additional insight provided by the brushing histograms strategy, and the potential to develop brushing histograms as an independent query component that could be embedded in various information systems.

Figure 4 shows the mean user satisfaction ratings for each query tool on 7 different survey questions. The pronounced significant difference occurs in Q5 (Helps to identify relationships among attributes?) with $p<0.05$. This result coincides well with the performance results on the more complex tasks. Users clearly recognize the additional insight provided by the brushing histograms. Since DQ sliders and brushing histograms use 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, no significant difference is observed for these questions. In addition, there was no statistical difference for Q7 (Overall reaction to the system?).

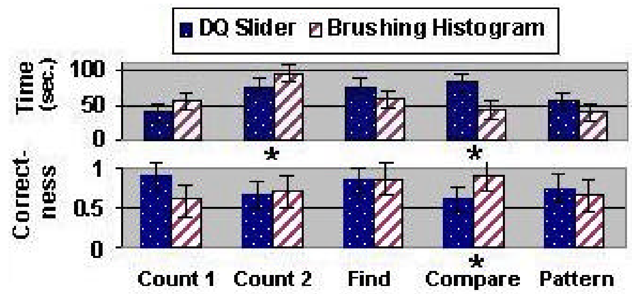

Figure 3: Mean user performance time and correctness for each task and query tool. Asterisks indicate significant difference at p<0.05.

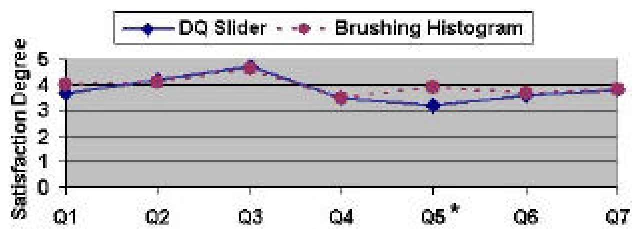

Figure 4: Mean user satisfaction ratings for each query tool. Asterisks indicate difference at p<0.05.

CONCLUSION

In summary, brushing histograms are superior for more complex discovery tasks, and more highly rated by users for relationship identification. This indicates its ability to function on its own as an information visualization tool. On the other hand, DQ sliders are superior for more simple range specification tasks, and function more as an auxiliary control for other visualizations. However, we noted that this advantage of DQ sliders 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. For example, brushing histograms could be zoomable (e.g. show data distribution within a sub-range), or have granularity controls (e.g. allow users to customize the granularity of the histogram bars).

REFERENCES

[1] Ahlberg, C., Williamson, C., Shneiderman, B. Dynamic Queries for Information Exploration: An Implementation and Evaluation, *ACM CHI'92*, pg. 619-626, (1992).

[2] Andrienko, G., Andrienko, N. Interactive maps for visual data exploration, *Intl. Journal of Geographical Information Science*, 13(4), 355-374, (1999).

[3] Dang, G., North, C., Shneiderman B. Dynamic Queries and Brushing on Choropleth Maps, *IEEE IV'01 Intl.*, (2001).

[4] Tweedie, L., Spence, B., Williams, D., Bhogal, R. The Attribute Explorer, *CHI '94 companion*, pg. 435-436, (1994).