Leveraging Interaction History for Intelligent Configuration of Multiple Coordinated Views in Visualization Tools

Andrew Pachuilo*
Texas A&M University

Eric D. Ragan†
Texas A&M University

John R. Goodall‡
Oak Ridge National Laboratory

ABSTRACT

Visualization tools can take advantage of multiple coordinated views to support analysis of large, multidimensional data sets. Effective design of such views and layouts can be challenging, but understanding users' analysis strategies can inform design improvements. We outline an approach for intelligent design configuration of visualization tools with multiple coordinated views, and we discuss a proposed software framework to support the approach. The proposed software framework could capture and learn from user interaction data to automate new compositions of views and widgets. Such a framework could reduce the time needed for meta analysis of the visualization use and lead to more effective visualization design.

Index Terms: H.5.2 [Information interfaces and presentation (e.g., HCI)]: User-centered design;

1 Introduction

Analysis of large, multidimensional data sets is challenging due to the need to inspect and interpret different attributes or data types concurrently. To address this challenge, visualization tools often provide multiple views that allow analysts different perspectives of the data [7]. With the application of common methods such as small-multiple views, focus-plus-context viewing, and brushing and linking, multiple views can be highly effective for allowing analysts to inspect different properties of complex data sets.

In practical scenarios, analysts tend to rely more heavily on reduced subsets of all available data, and different views might be preferred for different people or certain strategies. From a design standpoint, predicting what data attributes an analyst will find most useful can be challenging, and with high enough dimensionality, it is impractical to show all dimensions. Filtering data based on certain attributes can help speed up analysis by reducing the amount of data visualized at a time, but having enough widgets to adjust all possible data types or attributes would be unwieldy or impossible due to limitations in screen space. Additionally, certain data attributes might be understood more easily when viewed together, so it would be beneficial to keep such views in close proximity to make analysis more efficient.

With so many considerations, determining an appropriate configuration of views and widgets is a non-trivial task for the design of visual analysis tools. Researchers have proposed recommender systems to help select appropriate views based on properties of the data and common visualization guidelines (e.g., [4, 8]). However, for longer analysis sessions, it is important to consider user preferences and strategies. We propose an approach for improving the design and layout of multiple coordinated views by analyzing user interaction patterns collected through system logs. Once enough logs are collected they will be automatically processed and data mined to gain more insight about users and interpreted by an intelligent configuration process to create a new design. The recommended designs can be rolled out for use and additional log collection for iterative learning and design improvement. In our current work, we are considering this problem and approach for cyber security visualization tools.

*e-mail: apachuilo@tamu.edu
†e-mail:eragan@tamu.edu
‡e-mail:jgoodall@ornl.gov

2 Cyber Security Visualization Scenario

While our approach is not necessarily limited to any particular domain, we discuss our research of intelligent view configuration in the context of cyber security analysis. In our research, we work with cyber security analysts tasked with investigating and identifying of suspicious network activity. Cyber analysts must routinely monitor and sift through a large collection of data with numerous fields. Designing visualizations for such tasks can be difficult due to the exploratory nature of the task, the high volume of data, and the continuous streaming of incoming data. In addition to the raw data, cyber tools also often incorporate alerts from signature-based detection systems or analytic techniques (e.g., anomaly detection) to help flag potentially interesting or suspicious items. Consider a cyber security system that collects intrusion detection alerts from network activity across the globe. An analyst may wish to filter the data to only view intrusion alerts related to a specific source country over a specific port. To further simplify the task, they may opt to limit viewing to only those alerts flagged as the highest priority level by the system. As part of the analysis, the analysts might also be looking for recurring relationships over a certain period of time to better protect current systems, and these relationships could be easily spotted via certain filtered views of the data.

With such complex data and layered analysis goals, multiple views can be of assistance in helping analysts to make sense of different types of information together. At the same time, multiple widgets can help analysts to filter data, choose preferred visualization methods, or request additional data types for inspection. Previous tools have taken this approach. For example, the *Time-based Network traffic Visualizer* by Goodall et al. [2] providing a focused view on the packet level in the context of a network traffic view. In another example, Noel et al. [5] use multiple views to show network attack graphs, matrix representations, vulnerability details, and user annotations.

In our own work, we are also designing tools with multiple coordinated views. Figure 1 shows a screenshot of our cyber analysis prototype. The figure shows a collection of coordinated bar charts, histograms, and a map view to provide a composite view of cyber alerts. The views are interactive and double as widgets to filter or link selected attributes in other views. Having many multiple coordinated views and widgets together can be useful, but a designer's a priori view layout may not be optimal. Using this tool, we discuss a method for capturing and studying analysis patterns in order to improve the composition of views and the effectiveness of the visualization.

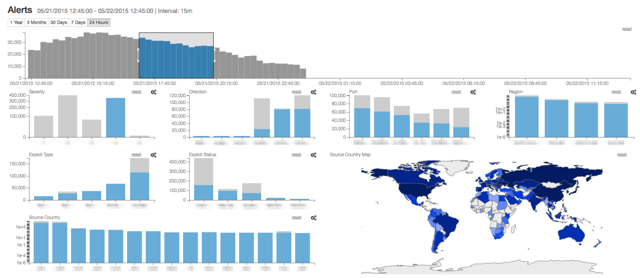

Figure 1: A cyber analysis prototype with multiple coordinated views.

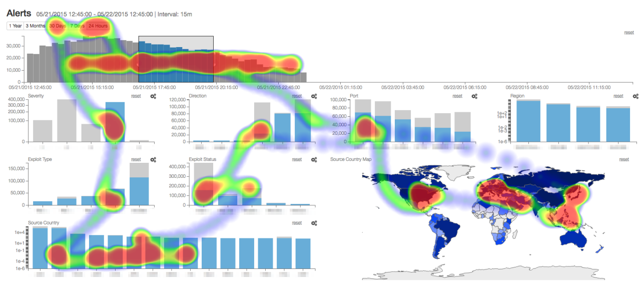

Figure 2: Interaction patterns (as shown here by a heatmap overlay) can reveal preferred views.

3 Intelligent Configuration of Multiple Views

To address the challenges in designing multi-view visualization tools, we discuss a general approach for improving the process of configuring views by capturing and data mining user interaction logs. Using this method, we can learn various strategies users take during analysis via their interactions and adopt them into a new configuration that better fits their needs. The methods requires a highly composable software framework that supports both logging of various interaction types and flexible configuration of the tool's visualizations and view layout.

The most useful interaction data to log will depend on the purpose for using the history data [6]. For our current cyber analysis scenario, we are most interested in eye tracking, mouse movements, keystrokes, and visualization meta-data (e.g., view state and data properties associated with interactions). Mouse and keyboard input demonstrates basic interaction history, eye tracking data provides a record of informational attention that may be independent of system input, and widget meta-data is important for matching input and viewing data with data state to learn analysis strategies.

Once a substantial amount of data has been captured, it can be mined to learn user priorities, usage patterns about how multiple widgets are used together, and which views are most beneficial for a given task. Additionally, interaction patterns could inform predictive models about probable sequences of actions. This information could then be interpreted to compose a new design that better streamlines user mouse movements, better groups certain widget or views, removes less useful items, and adds new views. Once the new design has been created, it will be immediately rolled out so that more information can be collected about the new design to provide richer insight for designing future iterations.

Implementation of such an approach will require a software framework that automates the processes of collecting interaction data, learning interaction patterns, and configuring views and widgets. The first step will be to create a framework for web-based analysis tasks for online testing to collect enough interaction logs for data mining. After the first iteration of collection, the data will undergo exploratory data mining with various methods to find what works best at understanding user strategies, movement, and the data set. Possible methods include using quality metrics [1] to better understand user strategies, exploring various machine learning algorithms for creating predictive models, and exploring methods for generating adaptive configurations for the layout [3] to better streamline interaction with the visualization.

This information can be used to create an intelligent system to interpret interaction data with consideration for specific users and particular data sets to create new layouts. The system would then roll out the new designs for use to gain additional interaction data for future iterations. With the generation of multiple layouts, it could also be valuable to find a way to map certain analysis strategies to different layouts so that particular layouts could be better optimized for certain tasks.

The automated framework should also include support for visualization tools to aide in the meta analysis of the framework itself. Determining what visualization designs would be most beneficial for meta analysis of interaction history would require additional research alongside exploring the various metrics and machine learning methods to use for data mining. These tools would help with understanding usage patterns for deciding possible metrics and machine learning approaches as a means to test the new designs for the framework to automatically generate. Figure 2 shows an example of a heatmap visualization tool that could help compare different layout configurations and visualization designs to help assess the effectiveness of the automation framework. Overlaying two heatmaps—one for eye movement and one for mouse interaction—for a certain time period of usage could also help researchers understand if a sensible decision was made during rearrangement of widgets. Another research goal for the development of the framework is to find a way to incorporate widget and view metadata into these tools to help make sense of the framework's recommended outcomes when considering user strategies.

Such a software framework that automates the configuration of multiple views could aide in the creation of more effective visualization design and reduce the amount of time spent manually doing meta analysis of the visualization. In future work, we plan on implementing and testing the proposed approach for creating an appropriate configuration for cyber analysis tools and multidimensional data.

Acknowledgements

This material is based on work supported by NSF 1565725. This manuscript has been authored, in part, by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. This research used resources of the Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DEAC0500OR22725.

References

[1] E. Bertini, A. Tatu, and D. Keim. Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2203–2212, Dec 2011.

[2] J. R. Goodall, W. G. Lutters, P. Rheingans, and A. Komlodi. Preserving the big picture: visual network traffic analysis with tnv. In *IEEE Workshop on Visualization for Computer Security, 2005. (VizSEC 05).*, pages 47–54, Oct 2005.

[3] P. Langley. Machine learning for adaptive user interfaces. In *Annual Conference on Artificial Intelligence*, pages 53–62. Springer, 1997.

[4] J. Mackinlay, P. Hanrahan, and C. Stolte. Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1137–1144, 2007.

[5] S. Noel, M. Jacobs, P. Kalapa, and S. Jajodia. Multiple coordinated views for network attack graphs. In *IEEE Workshop on Visualization for Computer Security, 2005. (VizSEC 05).*, pages 99–106, Oct 2005.

[6] E. D. Ragan, A. Endert, J. Sanyal, and J. Chen. Characterizing provenance in visualization and data analysis: An organizational framework of provenance types and purposes. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):31–40, Jan 2016.

[7] M. Q. Wang Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '00, pages 110–119, New York, NY, USA, 2000. ACM.

[8] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics*, 22(1):649–658, 2016.