Exploring the Use of Heuristics for Evaluation of an Immersive Analytic System

Todd V. Billow and Joseph A. Cottam Member, IEEE

ABSTRACT

Heuristics are a discount method for expert evaluation of systems. They help to identify usability problems in and can serve as guidelines for design and development, pending more formal testing and evaluation. As a new field, immersive analytics has not yet developed a set of common-practices or guidelines that might be coalesced into heuristics designed specifically for the field. This paper reviews the Nielsen usability heuristics as a starting point for the evaluation of immersive analytic systems. We present GeoVisor, an immersive analytic system in virtual reality for geographic visualization and abstract visualization of data, and use it as the target of our evaluation. We find the majority of the Nielsen heuristics provide useful insight into the issues we have encountered during GeoVisor’s development. Tellingly, some of our current issues in GeoVisor revolve around areas where Nielsen’s heuristics do not provide direct guidance. This indicates areas where these heuristics need to be modified to adapt to the development of immersive analytic systems, as well as areas for further research into abstract visualization of data in immersive environments.

Keywords: Virtual reality, Heuristic evaluations, Usability testing, Visualization, Visual analytics, Information visualization.

1 INTRODUCTION

Usability heuristics are a valuable tool for the evaluation of final systems and for guiding design decisions during the development process. Heuristics provide a method to improve system designs that is less expensive and less time-consuming than user testing and other usability research methods [2] (a so-called “discount method”). Mature heuristics can be applied informally by individuals as part of the design process, or in a more detailed formal evaluation by a group of experts.

Usability heuristics encode a particular point of view of a subject area. A set of heuristics (such as Nielsen’s) is an attempt to abstract a large set of design guidelines and common-practices into a smaller subset that is easier to track and evaluate. Therefore, there may be many different heuristics in a mature field. For example, usability heuristics for computer user interfaces have been provided by Nielson [7], Schneiderman [9], Stone et al. [10], and Johnson [4]. Because immersive analytics is an emerging field, there is not a well-defined set of guidelines from which to abstract heuristics that address the unique challenges of its design space. Although some attempts have been made to define heuristics for Information Visualizations and Virtual Environments, these do not address particular issues with abstract visualizations in 3D.

In this paper, we provide an overview of our work-in-progress, GeoVisor, an Immersive Analytics system in Virtual Reality (VR) (Section 2). We describe our findings from applying Nielsen’s heuristics to GeoVisor (Section 3) and identify areas where the heuristics were insufficient (Section 4). In this light, we discuss areas for future work in Immersive Analytics usability evaluation and propose a few topics that should be considered for any proposed heuristics in the domain (Section 5).

2 GEOVISOR

GeoVisor provides an immersive environment for viewing data visualizations that are situated in a geographic space (Figure 1). It has three view modes. The first is a standalone thematic map. This map may be augmented with additional data such as LIDAR depths or satellite images. The second is a standalone view of a single visualization (Figure 3), in which the visualization grows to fill more of the user’s view. The third view is a virtual workspace that bridges the other two. It shows visualizations of abstract data around the user on virtual “shelves”. The abstract visualizations and the map itself are coordinated through selection brushing and filtering.

Inputs. The majority of input is done with hand-held controllers. The majority of direct manipulation is done with the controller held in the dominant hand, while more abstract control (mode switching, menu selection, etc.) is performed using the controller held in the...
non-dominant hand. For interactions using two points of direct manipulation (e.g., zoom or rotate), the non-dominant hand controller temporarily assumes direct manipulation roles. Controller modes and menus are context-sensitive, largely driven by the object the user is interacting with. Tooltips provide constant hinting to aid the user in understanding how to interact with the system.

**Abstract Visualizations.** Our initial prototype allows users to choose a dataset and apply an appropriate visualization metaphor. Visualization options in 3D include node-link diagram (Figure 2), scatter plot, time series, parallel coordinates (Figure 4), and topic clusters. Users are able to perform selection brushing and linking techniques to highlight correlated data across visualizations. A “wand” is represented on the end of the tool controller to provide the user with a precise tip for selecting small objects such as nodes. The user is also able to control the rotational positioning of a visualization using the tool controller’s track pad, and in standalone view is able to adjust the scale as well.

**Map.** The “Geo” in GeoVisor is a centrally located map that rests at the height of the user’s lap when sitting (Figure 5). Since a primary use case for the system is to identify geospatial correlations in the data, the map is always visible. Using the menu controller the user is able to toggle various visualization layers, including simple location markers, a space/time plot, and LIDAR data. The user is also able to pan and zoom the map via the tool controller’s track pad.

**Timeline.** The timeline is an interactive time-series visualization that flies above the workspace and provides a method for filtering the visualizations based on time stamps in the data (Figure 6). The timeline tracks with the forward vector of the user’s head and follows the user so that it is always directly above and in front of the user. When the user’s head is pointed directly (gazing) at the timeline, the timeline activates two extent handles that the user is able to control remotely via the controllers. Once engaged, the user can filter the data by moving the controllers from side to side, controlling the extent handles remotely (indirect manipulation).

**Ergonomic Considerations.** Particular care has been given to design an environment for the ergonomic comfort of a user over long periods of time. Client feedback indicates sessions may last hours. The user will mainly be seated, though occasionally standing or making small lateral movements. To account for this, GeoVisor allows for both direct and indirect manipulation [8] of the virtual environment. Indirect manipulation is provided through hand-held controller trackpads and gaze focusing.

**Development Platform.** Our prototype system is an HTC Vive head-mounted display (HMD) and controllers connected to a Microsoft Windows desktop workstation. The software is developed...
on the Unity/Steam platform, integrating a handful of open-source toolkits such as VRTK [https://vrtoolkit.readme.io/].

3 A HEURISTIC EVALUATION OF GEOVISOR

Heuristic evaluations are for inspecting software for usability problems. Many heuristic evaluations are formally performed by a group of expert evaluators. But, because GeoVisor is in an early design and prototyping phase, we have only performed an informal evaluation. The advantage of informal evaluation is that it enables rapid, iterative development while still being guided by well-founded principles. In some ways, informal heuristic evaluation is like unit testing: it can reveal the presence of problems, but does not prove their absence. Using informal heuristic evaluation has enabled us to find some problems that could be corrected as part of short prototyping cycles with minimal investment.

Nielsen’s heuristics for usability started with a collaboration with Molich [6]. These were later refined through evaluation of 249 problems [7]. They have been widely accepted in HCI research and are frequently referenced in literature. Our team had prior experience with Nielsen’s heuristics in other contexts, and chose to apply them to GeoVisor. Our immediate goal was to discover issues with GeoVisor. The results are instructive for evaluation of other systems and can guide future heuristic design.

Here we describe Nielsen’s heuristics and discuss our findings, having applied them to GeoVisor. In the majority of cases, we applied the heuristics to a functioning prototype but in a few cases (noted in context) we applied the heuristics to a proposed design instead.

Visibility of System Status: The system should keep the user informed of its state in a timely and contextually appropriate way.

GeoVisor succeeds in providing system status for various operational modes. Its virtual representations of the user’s hand controllers contextually change their display depending on the system state. Data instantiated in the workspace is represented as a cylinder when there is no visualization attached. Data selected in a visualization is represented with a highlight color and any corresponding data in other visualizations is also represented. Loading indicators will be displayed whenever the system is unable to respond immediately to user requests.

Match between system and real world: The interface should match the user’s pre-existing context. This includes using terms, iconography and ordering that are already familiar to them.

GeoVisor follows real-world conventions in its simple, natural language toward the user in tips and menus. Interactions with the map are analogous to the direct manipulation interaction used in many touch-based map interfaces. This heuristic was otherwise difficult to apply because much of the interface and visualizations are abstracted from the real world. Developers of immersive environments that attempt to emulate the real world would likely find this heuristic more applicable.

User control and freedom: Support undo/redo so the user can confidently enter and easily exit new states. This allows the user to control the direction of a transformation and affords the freedom to explore unfamiliar areas.

GeoVisor allows for users to rearrange the workspace and provides flexibility for different visualizations to be instantiated from datasets. We identified a problem in that the current design does not support undo or redo actions. We intend to revisit this as we harden the software in the future.

Consistency and standards: Eliminate incidental differences. The user should not need to wonder if different appearances imply different states or are just variations on what has been seen before.

We found this to be a problematic heuristic for GeoVisor as a VR application. The design does maintain consistency in input interactions across modes (where possible), and it uses conventions where there might be similarity to desktop or mobile systems (the workspace visualization containers are analogous to desktop windows with actions such as new, open, close, etc.). Jerald’s “The VR Book” [3] is a good source for interaction design guidelines for VR. However standardization of devices and inputs has not yet been achieved in general as VR technologies are changing at a rapid pace due to recent industry competition. Though progress has been made since the survey published by Bowman et al. [1], this lack of standardization continues to make it difficult to apply platform conventions per Nielsen’s recommendation.

Error prevention: Structure interactions so errors are avoided. This includes validating inputs, guiding interactions away from error states and designing systems with few error states.

GeoVisor directly prevents many error states. For example, a user cannot instantiate a visualization without first selecting a data source. Also, controllers are modal and only provide methods for the user to perform actions that are allowed in the current mode. Confirmation dialogs are used for permanent actions, such as delete. In our early prototype, we identified a difficulty in selecting small visualization objects when the user is interacting with multiple views in the workspace. This leads to selection errors (both selecting objects unintentionally and not being able to select the desired objects). We corrected this by limiting the user to selection of data objects only when viewing a standalone visualization in standalone view mode. We also identified problems with reaching some of the buttons on the HTC Vive controllers, in that some buttons were difficult to access (the menu button above the touchpad) or difficult to actuate (the side “grip” buttons on the handles). We reconfigured the interactions to avoid the use of these buttons.

Recognition rather than recall: Make possible actions visible to a user so they can recognize what to do rather than recall how to do it.

While GeoVisor provides affordances for interactions—making contextual menus, selection indicators, and tools visible to the user—the heuristic did expose a problem in our workspace view of multiple visualizations. We did not label each visualization with the name of its underlying dataset. This forced the user to remember which data set was selected, taxing the user’s working memory.

Flexibility and efficiency of use: Hidden accelerators (such as keyboard shortcuts) allow expertise to impact efficiency without being detrimental to novices. Enabling users to tailor shortcuts for frequent actions is also beneficial.

This heuristic exposed a design choice we made because of the development goals of the project. GeoVisor, in this first development phase, is intended as a proof-of-concept. We did not intend to create a system with advanced features for expert users. Rather, we wanted to first prove its usefulness in exploring data with a geographic map as a base in VR. This heuristic can serve as a foundation for possible future interactions that aid expert users or allow them to enhance and customize the system via scripting or an API.
Aesthetic and minimalist design: Excessive elements diminish the visibility of essential ones. Therefore, irrelevant or rarely needed information should not be prominently featured.

Some initial design features of the workspace incorporated a semi-transparent cube drawn around a visualization when selected by the user. As we evaluated for this heuristic, we recognized that this proved to be an unnecessary and distracting element in the prototype. We reduced the visual clutter caused by this element by creating a highlighted state for the base of the visualization container, reducing occlusion of the visualization itself while still indicating its active state. The lighting of the scene also proved to be a problem, as our early prototypes included directional lighting that cast shadows and material properties of environmental objects that created specular highlights that were distracting. We removed the environmental lighting and created materials that were self-illuminating, and added environmental fog to reduce visual clutter caused by the map's visibility to the extents of the scene.

Help users recognize, diagnose and recover from errors:
Expressing error information in plain language is preferable to error codes. Include a description of the problem and possible resolution steps.

As with the reasons listed in the above flexibility and efficiency of use heuristic, we had not yet created a systematic design for error messages. As the software matures, we intend to add these messages where appropriate and then re-evaluate the system against this heuristic.

Help and documentation: Large systems often require external sources of information. This information should be user-task focused, include concrete steps and not be overly large.

This heuristic revealed that our designs for controllers and environmental objects did not include any tooltips or hints for the user. We created a standard for incorporating tooltips for every controller interaction as well as hinting on the controller touch-pads via arrow icons and text. We also identified the need for documentation of system actions and capabilities to be compiled as the software reaches maturity for end-user delivery.

4 Discussion

Our informal evaluation of GeoVisor using Nielsen's heuristics enabled us to discover and correct a number of usability problems in GeoVisor. As the software matures further, a formal evaluation with multiple experts coupled with user studies is desirable for identifying more of the probable usability problems in the system.

A number of the usability problems we discovered during our evaluation were not necessarily related to the immersive nature of the system. Nielsen proved valuable in revealing usability problems that are common to desktop, mobile, and immersive environments. However, as we have developed GeoVisor, we have also encountered some usability considerations that were not adequately framed by Nielsen's heuristics.

One of the problems we identified is due to the immaturity of current Virtual Reality technology. The HTC Vive's display, similar to other current technology, is able to render its display with enough frequency to give the user a sense of immersion. It does this at the cost of display resolution, a problem commonly known as the "screen door" effect since the user is able to see the individual pixels of the relatively low resolution display, creating the appearance of a wire screen in the view. This causes significant problems with rendering text at small sizes. There is little that immersive visualization system designers can do to alleviate this, but there are many institutions working on this problem.

We also identified potential issues caused by the field of view available to the user in the immersive environment. In GeoVisor we've given users the ability to instantiate visualizations "in the round" (i.e., on all lateral sides relative to the user). This allows for the possibility that a data object highlighted in the view of the user might also cause a correlated data object to highlight in a visualization that is not simultaneously viewable. Further research is needed to explore unobtrusive ways to indicate system status to make the user aware of changes that are outside of the field of view. This is analogous to brushing and linking on infinite canvases in 2D, but there is likely a distinct solution for immersive environments.

Another problem we discovered was related to immersion. Our initial designs for GeoVisor brought visualization objects very near to the user's personal space (to support ergonomics) but this caused psychological discomfort as the space became crowded. We resolved this by moving the workspace containers further from the user's point of view. This incurs a cost in ergonomics, but provides a net benefit of comfort using the system. This illustrates a conflict between ergonomic and immersive considerations. Providing user-tunable parameters for visual scale, spacing and reach would help resolve these issues. Providing this reconfigurability in the ergonomic space may apply to Nielsen's heuristic of flexibility and efficiency of use, but its importance in immersive environments may not be immediately apparent to evaluators.

Also related to immersion, Nielsen's heuristics do not address other sensory inputs such as haptic and auditory feedback. While the absence of these may not be problematic to usability, their presence may significantly enhance the user's ability to sense objects and cues in the immersed environment. In immersive analytics in particular, there may be some benefit to sensing relative values via sound or touch differences (imagine a bar chart with haptic feedback that changes as you touch different heights of bars, or nodes that make different sounds depending on their properties).

Lastly, Nielsen's heuristics do not address the usability of data visualizations per se. For example, they do not directly address many issues related to accurately interpreting data. Since the primary task for users of GeoVisor is to interact with its visualizations to analyze data, this represents a significant gap in these heuristics as applied to GeoVisor and potentially for other immersive analytic systems.

5 Future Work

A number of heuristic sets have been proposed for evaluation of information visualization, including a set proposed by Forsell and Johnson [14] which resulted from an analysis of six prior proposals. Designers of immersive analytic systems may find these heuristics helpful for evaluation of immersive data visualization. However, these heuristics do not directly address specific questions arising from 3D abstract representation of data in immersive environments, such as depth-perception and freedom of movement. Ware addresses some of these issues in his book "Information Visualization: Perception for Design" [13]. Luboschik et al. [5] also address the question of spatial perception issues in Augmented Reality, but there appears to be a paucity of literature addressing these questions.

Sutcliffe and Gault [11] have proposed a list of heuristics for Virtual Environments based on Nielsen's usability heuristics, but many of these are based on the assumption that the Virtual Environment should "represent the real-world as faithfully as possible." They suggest that Virtual Environments that represent unnatural worlds would require a set of heuristics that fit the user's model of the task and domain. For example, a real-world emulation might include lighting artifacts such as material reflection properties and shadows that may interfere with a user's understanding of data. Therefore, designers of immersive analytic systems who attempt to synthesize a set of heuristics from those listed above will have to contend with the competing priorities of real-world emulation and the abstract visualization of data.
Weidert, and Troy Zuroske. We have presented GeoVisor, an immersive analytic system in virtual visualization. Though heuristic evaluation is only one of many methods for evaluation of software systems [12], we contend that heuristic evaluation continues to offer significant value to system designers of immersive analytic environments as an inexpensive approach to identifying usability problems. However, while there has been some effort to define a set of heuristics for information visualization evaluation, there are no formally defined heuristics for abstract 3D data visualization in immersive environments. In conclusion, there is a strong need for research toward understanding the benefits and best practices of visualizing data abstractly in three dimensions for human reasoning and analysis in these environments. Designers of immersive analytic systems would benefit significantly from a set of heuristics designed specifically for this domain.

**Acknowledgments**

The authors wish to thank Russ Burtner, Nick Cramer, Scott Dowson, Nathan Hilliard, Dave Lewis, Sierra Maple, Rick Riensche, Paul Weidert, and Troy Zuroske.

**References**


