Tangible Interaction with the R Software Environment Using the Meuse Dataset

Rachel Bradford, Landon Rogge, Dr. Brygg Ullmer, Dr. Christopher White

Abstract

The objective of this project is to interpret large quantities of information through the use of tangible interfaces, in such a way that scientists and the general public can comprehend and ask questions of complex data sets. We have attempted to realize this objective by modifying and adapting existing tangible interaction systems to perform new functions. Specifically, we used the Apple iPad and the Microsoft Surface as test platforms, and tangibles as physical interactive representations of digital data.

Interaction Concept

The general approach guiding this research was to employ a tangible interface for the purpose of controlling graphical visualizations of large datasets.

The iPad device provides form factor well-suited to single user interaction. This allows the user to engage with a variety of audience sizes. The output visualizations appear on a separate monitor or projection screen depending on the target audience. As a future possibility, in which audiences could each have an iPad and collectively interact with the dataset. Scaling the interaction to this level will require more research into how parameters can be changed by multiple users simultaneously.

The Microsoft Surface allows a smaller group of users to directly engage with the dataset. This interface could be optimal for small collaborative groups. Visualizations would appear either directly on the surface or on a separate display.

R and Meuse Context

We have chosen the Meuse Dataset because it is a well developed R training dataset with properties resembling several other domains of interests (e.g. petroleum engineering, environmental remediation). Using the dataset with the R software environment provides ready access to visualizations of a variety of parameters relating to environmental impact of past industrial activity (e.g. concentrations of cadmium and zinc). In addition, visualizations can be made relating to flood frequency, elevation, organic matter loss and distance from the riverbed. The R environment supports numerical and graphical analysis of this dataset.

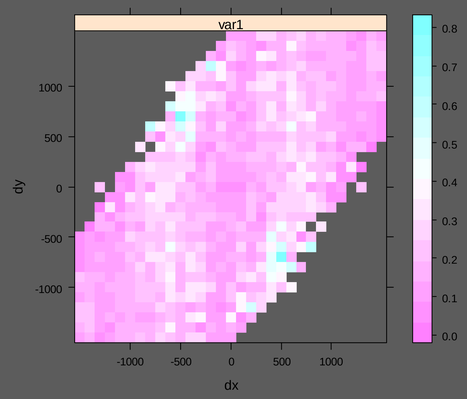


Fig. 1 - R visualization of a variance graph which describes how far data points lie from the mean.

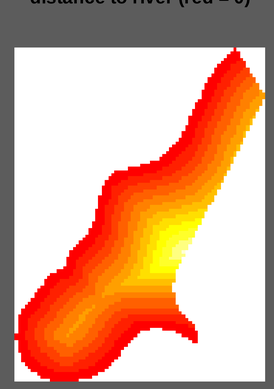


Fig. 2 – R visualization of a distance graph where red equals zero.

Fig. 3 - The iPad surface once all tangible interface elements are in place.

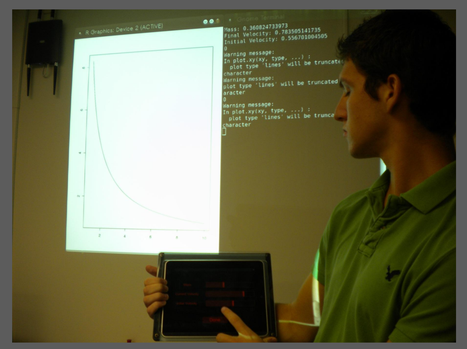


Fig. 4 - The user inputs parameters on iPad resulting in a graphical visualization on the display.

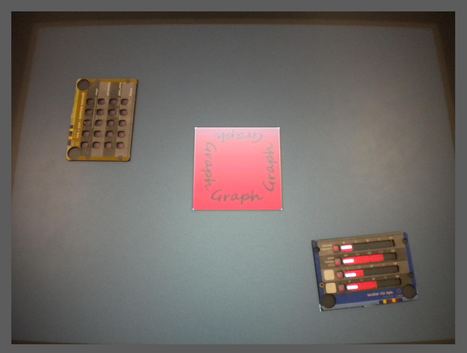
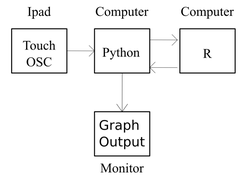


Fig. 5 - Microsoft Surface with interaction devices on the surface. Sliders and buttons inside the devices are loaded dynamically by the Surface.

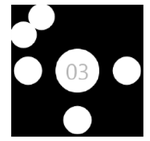
Fig. 6 - Multiple users interacting with the tangible elements on the Microsoft Surface.

Interaction Across Systems

For the iPad interaction system we have used the Python programming language to receive data from the TouchOSC app on the iPad (iOS application for parameter control). We then use Python to send this data to the R programming language, which creates graphs of the data (See Figs. 1-4). Our plan is to be able to equip the iPad with RFID (Radio Frequency Identification) readers, which would receive the ID of the interaction devices, and load the appropriate content.



iPad Control Flow Diagram



Microsoft Surface Byte Tag

For the Microsoft Surface, we utilized the built-in visual tag recognition system to recognize the interaction devices. The system detects and identifies the tags, and uses the identification number associated with each tag to load content which is functionally equivalent to that displayed on the iPad. In order to control the tag and gesture recognition libraries we used the C# programming language and XAML descriptions of our content.

Device Self-Awareness

In developing this system, we have studied the concept of creating interaction devices which are self-aware. This means that each device contains the necessary elements to identify itself to the system with which it is interacting and to give the system key information about itself and specific details about its functionality.

Our software for the Microsoft Surface implementation has already begun to engage self-awareness properties (introspection/reflection) in its handling of visual tags. We have also begun similar code for the iPad using RFID.

Acknowledgments

This project was sponsored by Louisiana State University's Center for Computation and Technology REU program. Funding was provided by the Louisiana Board of Regents and the National Science Foundation.