Bugarium: 3D Interaction for Supporting Large-Scale Bug Repositories Analysis

Papon Yongpisanpop, Hideaki Hata, Kenichi Matsumoto
Graduate School of Information Science
Nara Institute of Science and Technology, Japan
{papon-y, hata, matumoto}@is.naist.jp

ABSTRACT
Big data became problems not just how to analyze and visualize but also how to interact with the data. In software analysis and maintenance, bug tracking system receive feedbacks of the software project users everyday, which means that the data is increasing everyday. A large-scale bug tracking system that contains large amount of information does not give end users an easy way to analyze bug information because it lacks of good interaction system. We present Bugarium that integrate 3D Motion Controller and data-driven documents to ease both interaction and visualization on a large-scale bug repository. Bugarium leads to a significant increase in terms of using 3D motion controller to operate big data in software visualization. An user study shows that Bugarium made users satisfied while using it to interact and visualize a large-scale bug tracking system.

Categories and Subject Descriptors
H.4 [Information Systems Applications]: Miscellaneous; D.2.8 [Software Engineering]: [complexity measures, performance measures]

General Terms
Experimentation, Design, Measurement

Keywords
3D Interaction, software visualization, bug tracking, big data

1. INTRODUCTION
Bug tracking system is a software application that is designed to help keep track of reported software bugs. Bug tracking systems play an important role in software development [7, 8] in these days. They are widely used by open source software projects and industrial software development organizations. It allows developers, testers and end users to provide feedback of an incorrect or error or as a request for enhancement of software. In the other hand, bug tracking systems are also used to analyze and maintain software after release.

Having a bug tracking system is extremely valuable in software development. According to the research [3], for a large-scale, widely-used software system, it could receive a large number of bug reports. In 2010, Eclipse project received 49,422 bug reports which mean that the bugs were reported around 135 bugs everyday.

Big data can be turned into big insights. But most of the time it causes complexity and confusion in visualization. Visualizing a large amount of bug information has always been a big challenge in software engineering. Many researches [1, 3, 6, 8] have tried to solve this visualization problem by developing many tools such as bugmap [3], SourceVis [2], bug database [6], evolution history [5], and vocabulary [1].

According to the research [9, 11], using only mouse and keyboard slow sometimes users down while navigating through a large-scale data system. we realized that to perfectly visualize data we need both good data representation and good interaction method.

In this paper, we propose Bugarium, which is a tool that implements 3D motion controller that allows users to naturally use hands and fingers to interact with data in large-scale bug tracking systems.

2. THE STRUGGLE TO INTERACT WITH LARGE-SCALE DATA
The primary challenge in our research is to ease interaction of a visualization-based data discovery tool and seeks to derive more value from large-scale data.

Typically, large-Scale data systems use text-based style to represent the data such as Bugzilla, a bug tracking system that has been using in many open source software projects. The major component of Bugzilla is the database that records information about known bugs such as timestamp, severity, priority, erroneous program behavior, and details on how to reproduce bugs.

Normal interface of Bugzilla represents bug information in text-based style as shown on the left of Figure 1. Many researches that we already mentioned in Section 1 only focused on how to make meaning out of a big data by developed visualization-based data discovery tool.

In Figure 1 on the right, we show the relationships between 1197 committers and 1599 bug reports of Eclipse JDT core project in 2008 using force-directed graph which is one of the most famous visualizing graph to represent multi-relations data structure on a 30 inches display, which has the resolution of 2560x1600. Even we have a large display and new
visualization technique to represent large-scale data but yet we still facing the problem that all data cannot fit in the display and it is hard to navigate through the data by using only mouse.

3. **BUGARIUM**

3.1 **System Architecture**

Figure 2 presents an overview of Bugarium architecture. The system is composed of three layers: view, controller and model. Each layer serves different purposes and works independently from each other.

The top layer is view layer. The purpose of this layer is to display outputs and receive motion inputs from user. Leap Motion is a small USB peripheral device, which is designed to be placed on a physical desktop, facing upward. Using two monochromatic IR cameras and three infrared LEDs, the device observes a roughly hemispherical area. The LEDs generate a 3D pattern of dots of IR light and the cameras generate frames of reflected data, which is then sent to the computer. The motions are analyzed by the Leap Motion controller software.

The middle layer is controller. It has two parallel components: 3D motion interpreter and data controller run together.

Once user interacts with the data in view layer, user interface will pass 3D motion data from Leap Motion to 3D motion interpreter to interpret and send it to data controller. Data controller communicate with data layer to get provided JSON data structure that fed from Bugzilla. Data controller uses D3.js, which is a JavaScript library for manipulating documents based on data to bring data back to user with powerful visualization components.

3.2 **User Interaction with LEAP MOTION**

We tried to replace mouse by using motion interaction. Bugarium allows users to swipe, zoom-in and out, select, grab or even using the basic hand signs to interact with the data. In this paper we show two interactions, which user can categorize bug reports by selecting priority of the reports using hand signs as shown in Figure 3 and selection comparison for bug reports using fingers shown in Figure 4.

In Figure 3 on the left shows the big picture of the relationships between Eclipse JDT core bug reports and committers in 2008 which have more than 3000 nodes liking together. What if you would like to view only bug reports that have priority $= 5$? The problem is even Bugarium has shown a proper graph to represent the multiple relation in bugs but it is not easy to visualize at all when having such a big data.

Normally to select bug reports based on the priority using Bugzilla, users need to click on the selection box to select the level of priority. Or even other visualizing tools need to provide interface option for users to use a mouse to click on it. But Bugarium allows you to use just the basic hand signs to select bug reports as shown in Figure 3 based on the priority without any additional interface component such as checkbox or drop down menu.

Figure 4 shows the benefit of using two hands and fingers to select 2 bug reports for the comparison, which using mouse can not do something like this.

4. **AN EXPERIMENT ON ECLIPSE**

We sampled 40,000 bug reports from Eclipse JDT core project from 2001 to 2008 and use them as an example to illustrate the usage of our tool. We set up experiments to evaluate the satisfaction of Bugarium overall usage. We selected ten participants from our software engineering lab., which have variety roles in software development process including programmers, testers and end users to use Bugarium to visualize and navigate through the bug reports.
5. RELATED WORK

In the past, many software visualization techniques have been proposed. For example, Kuhn et al. proposed a tool [1], which in the position of a software artifact reflects its vocabulary, and distance corresponds to similarity of vocabularies. D’Ambros et al. [6] proposed a system to visualize a bug database by using radiography technique to display bug information in the system level and introduced a “Bug Watch” to visualize a specific bug. Wettel et al. proposed Software Systems as Cities [10], which represents large-scale software systems as a city metaphor.

In 2002 the movie called Minority Report, in the scene that Tom Cruise manipulates data on a series of large screens with hands has inspired researchers that visualization is not just the way to represent the data but also how we interact and manipulate it. Computer scientist started to develop new type of user interface and input devices that not only existing in science fiction but now in reality [9].

Another research area that tries to use hand gestures is in medical field. Jacob et al. [4] proposed a technique to collaborate with a robotic scrub nurse which allow surgeons use hand gestures and/or voice commands without interrupting the natural flow of a procedure to command robot hand to pass instruments, sutures, and sponges during surgery.

6. DISCUSSIONS
Remote collaboration We would like to enable remote collaboration in Bugarium for some analysis tasks that users need to collaborate together.

Input technology We think that keyboard is not going to get replaced for a really long time, because it is really good at text. But we would like to provide another text input option by using microphone and voice recognition in Bugarium.

Data representation We will make the most meaning out of large-scale data visualization by research more on visualize technique to represent the specific data structure and appropriate motions that can be used to interact with the data.

8. ACKNOWLEDGMENTS

This study has been supported by JSPS KAKENHI Grant Number 25880015.

9. REFERENCES


