

A Methodological Evaluation of Natural User Interfaces for Immersive 3D Graph Explorations

Ugo Erra^a, Delfina Malandrino^b, Luca Pepe^b, Vittorio Scarano^b

^a*Dipartimento di Matematica, Informatica ed Economia, University of Basilicata, Viale dell'Ateneo, Potenza, ITALY*

^b*Dipartimento di Informatica, University of Salerno, Via Giovanni Paolo II, Fisciano (SA), ITALY*

Abstract

In this paper, we present a novel approach for a real-time 3D exploration and interaction of large graphs using an immersive virtual reality environment and a natural user interface. The implementation of the approach has been developed as plug-in module, named 3D Graph Explorer, for Gephi, an open software for graph and network analysis. To assess the validity of the approach and of the overall environment, we have also conducted an empirical evaluation study by grouping people in two different configurations to explore and interact with a large graph. Specifically, we designed an innovative configuration, exploiting the natural user interface in a virtual reality environment, against a well-known and widespread mouse-keyboard configuration. The evaluation suggests that these upcoming technologies are more challenging than the traditional ones, but enable user to be more involved during graph interaction and visualization tasks, given the enjoyable experience elicited when combining gestures-based interfaces and virtual reality.

Keywords: Software visualization, Natural User Interface, Virtual Reality, Graph Exploration, Evaluation Study

Email addresses: ugo.erra@unibas.it (Ugo Erra), delmal@dia.unisa.it (Delfina Malandrino), lucapepe@hotmail.it (Luca Pepe), vitsca@dia.unisa.it (Vittorio Scarano)

1. Introduction

A graph is a mathematical structure used to model pairwise relations between objects. Usually, the objects are drawn as nodes and relations are drawn as edges between nodes. This pictorial representation of the nodes and edges of a graph provides a visualization of these relationships that helps to understand the overall structures of complex systems. Recent interest in social networks, software architectures, planning and scheduling has led to the application of graph visualization and exploration techniques that provide analysts with relevant visual cues in order to understand intrinsic data structures. In particular, the visualization of large graphs has been empirically proven to be a tool that greatly enhances the ability to understand and characterize the relations between objects [1].

Graph drawing addresses the problem of constructing an automatic visualization of graphs using an algorithm. The goal is to derive an aesthetically-pleasing picture that follows the layout conventions of a given application domain. The annotated bibliography prepared in [2] discusses hundreds of papers that study layout algorithms, while various frameworks have been developed that implements such algorithms. As for instance, several tools are available for the representation of graphs in the plane such as Gephi [3], and the most of them also allow some 3D representation and managing, such as GIOTTO3D [4], 3DCube [5], and Gem3D [6]. In addition, modern off-the-shelf graphics processor units enables methods to draw large graphs in 3D space that can be used for information visualization. In contrast, very little research has addressed the problem of real-time 3D exploration and interaction with large graphs. In particular, when there is a large quantity of graph data to be visualized the arrangement of nodes and relations can affect understandability, usability and aesthetics, typically leading to information overload. In addition, exploration can be very difficult due to the inability to interact with it. Ideally users would be able to explore and interact with the graph in real-time 3D; they could observe it from different points of view, move one or more nodes, or group unimportant nodes into clusters in order to reduce information overload.

Research into virtual reality, used as a tool for the visualization of, and interaction with computer-simulated environments that immerse users in a real or imagined world, is well-established. Although this technology was first developed in the 1970s, it has only recently become widely available, through low-cost devices such as the Oculus Rift virtual reality headset [7]. In most

virtual reality applications, user interaction is based on input devices such as a keyboard, mouse, or joystick. However, this unnatural way of interacting with virtual objects breaks the illusion that users can directly interact with the virtual world. Recently, researchers have begun to explore gesture-based interaction with virtual reality content by using contactless motion-sensing devices. These devices, such as Microsoft Kinect [8] or Leap Motion [9], track the user’s body and hands in the physical space and enable developers to design invisible interfaces also called natural user interfaces. This enables the use of natural, hands-free gestures that deepen the immersion in a virtual reality application.

Actually, using a 3D visualization on a computer adds a well know range of difficulties, such as: the screen and the mouse are both 2D devices, so we perceive only a 3D projection on a 2D plane. It is difficult to control a 3D visualization with the interaction techniques that are currently in common use since they were designed for 2D manipulation (e.g., dragging, scrolling). Users need to pay attention to the navigation of the 3D view in addition to the navigation of the underlying model: the extra controls for navigation, zooming, etc. get in the way of the user’s primary task. Hence, although there are good 3D graph layout algorithms, they are not fully exploited due to limit of 2D visualization devices. In addition, also the adoption of a virtual reality headset that enables stereoscopic visualization of 3D graph raises the question of using a device that implement the interaction techniques more natural, efficient and appropriate in a virtual environment such as the natural user interface.

This paper describes a novel approach to 3D real-time interaction with large graphs, based on a natural user interface and immersive virtual reality. The approach has been implemented as a plug-in, named 3D Graph Explorer, for Gephi (open-source graph and network analysis software). It was empirically tested in an experiment that compared two configurations for the visualization, exploration, and interaction with a large graph. In brief, our contributions are: 1) A real-time 3D exploration and interaction system for graph visualization. 2) A natural user interface for 3D interaction in a virtual reality environment. 3) An empirical comparison of the proposed system with the traditional mouse-keyboard combination.

The remainder of the paper is structured as follows. Section 2 provides an overview of related work; it addresses other solutions proposed for 3D real-time interaction and manipulation in visualizing large 3D graphs. Section 3 describes the 3D Graph Explorer plug-in and presents the interaction design.

Section 4 presents the software architecture of the plug-in. The experimental comparison of the Oculus Rift/ Leap Motion configuration with a traditional mouse/ keyboard configuration is presented in Section 5, followed by the results (Section 6). We end with some final remarks and future directions for our research in Section 7.

2. Related Work

Currently, there are many open-source software tools available for large graph visualization and manipulation that can be extended via plug-ins. For example, although Cytoscape [10] was initially focused on biomolecular networks such as protein-protein or protein-DNA interactions, beginning with version 3.0, it has been generalized and can be considered as a general-purpose tool. Cytoscape supports the development of plug-ins using an ad-hoc Java-based framework and plug-ins are made available to the public. In Cytoscape, the graph is displayed in a 2D space and the user can interact with it via a GUI composed of a toolbar and a form. Various actions are allowed such as filtering and query-based node highlighting. The toolbar also allows network exploration. The user can use the zoom tool in order to focus on a region of the graph that has been previously selected with the mouse.

Another popular software tool is the Tulip Visualization Framework (Tulip)[11]. Tulip is written in *C++* and uses OpenGL for graph rendering. Like Cytoscape, Tulip allows various actions such as filtering, grouping, etc. With regard to graph exploration, Tulip offers more tools than Cytoscape. In Tulip the graph is displayed in a 3D space and the user can rotate it, and pan and zoom on a specific region. With regard to extensibility, Tulip only allows the development of algorithms and tools (e.g, metrics, exporter, layout) that interact with the graph currently displayed.

GDDToolKit [12] is a Graph Drawing Toolkit designed to efficiently manipulate several types of graph, and to automatically draw them according to many different aesthetic criteria and constraints. A graph is drawn using a sequence of logical steps, each step can be chosen from several algorithms. In such a way, the designer can trade between efficiency and effectiveness.

The Open Graph Drawing Framework [12] is a *C++* library of algorithms and data structures for graph drawing. The library offers a wide range of graph drawing algorithms that allow to reuse and replace particular algorithm phases by using a dedicated module mechanism. In such a way, these

algorithms can be used as building blocks of graph drawing algorithms and thus also providing a powerful platform for implementing new algorithms.

These interaction paradigms are performed using the modal interaction where the control of a graphical window is subordinate to an application's main window [13]. This means that a generic action requires at least two steps: (i) click a specific button that is located outside the region in which the graph is displayed (*graph region*), and then (ii) move the mouse cursor to the *graph region* to execute the action desired. This two-step paradigm has a disadvantage: the user is forced to go in and out of the *graph region* many times, limiting interactivity. This problem is avoided in 3D Graph Explorer as actions take place directly in the virtual environment, without using a toolbar or menu.

There are other issues with such software. Usually the point of view is fixed and the main actions are panning and zooming, and in some cases graph rotation. These actions are often mutually exclusive due to the use of the toolbar. Furthermore, none of the existing softwares takes the third dimension into full account. In practice, the user can only move nodes on a bidimensional plane. This failure to use the third dimension limits the layout and the potential to explore large graphs. Our work addresses these problems. 3D Graph Explorer uses a first-person camera control scheme. This control system is widely used in video games in order to navigate large 3D environments and guarantees high interactivity. Further details are given in Section 3.1.

With respect to input-output devices, existing graph visualization tools are mainly mouse-based and little work has addressed interactions using hands-free gestures. Nancel et al. in [14] studied the effectiveness of hands-free gesture interaction for panning and zooming on very large displays. These authors showed that hands-free interactions are less effective than traditional (mouse-based) interactions when high precision is required, due to a lack of guidance. In the non-high-precision context, hands-free interactions have been tested for medical image visualization and navigation [15][16]. Other areas in which natural user interfaces have been investigated are the fields of domotics [17] and robotics [18].

Graph exploration and manipulation are tasks that do not require high precision. To the best of our knowledge, there is no work in the current literature that has evaluated a natural user interface in this context. Our work fills this gap by providing an assessment of user engagement with these new types of interaction.

3. 3D Graph Explorer

3D Graph Explorer is built on Gephi [3] open-source software for exploring and manipulating networks. Our plug-in integrates with existing Gephi functionality and exploits its features. In particular, Gephi enables the layout of networks, the calculation of metrics for network nodes, the creation of clusters, and the ability to adjust the visual properties of the network. Some of these built-in layout algorithms also make possible to arrange nodes in a 3D space, although visualization and interaction is limited. More specifically, although the graph is arranged as a 3D model there is no way to rotate the viewpoint or move freely because the standard visualization only maintains a fixed camera in a 2D space. The only feasible real-time interactions are the pan/ zoom facility available in the toolbar, or the ability to move a node using the drag-and-drop feature.

The 3D Graph Explorer plug-in exploits Gephi's layout algorithms and, in addition, provides the following features:

- Visualization, exploration, and manipulation of graphs in the 3D space.
- Mouse/keyboard and natural user interface support for interactive information seeking.
- Virtual reality support for an immersive experience.
- Exploitation of Gephi features, such as layout algorithms and graph filters.

The plug-in also supports the exploration of graphs using the Visual Information Seeking Mantra, namely: overview, zoom and filter, and details-on-demand [19]:

- *Overview.* The 3D environment supports the visualization of the entire graph. The overview provides the user with an interactive visualization of the layout. The name of each node is attached as a label, which is displayed alongside the node. These labels can be shown or hidden as required.
- *Zoom and filter.* The interface allows the user to zoom in and out. For example, if the user identifies a group of nodes of interest in the overview, this group can be zoomed, selected, and analyzed. The zoom

and filter feature is particularly useful in the case of large graphs. However, because filters are complex operations that are wrapped into queries, which can be chained and combined, their use in context of natural user interface will not be discussed in this paper.

- *Details-on-demand.* A user can select one node and the associated properties are shown as required.

Figures 1 and 2 illustrate the 3D Graph Explorer in action. Note that the plug-in enable to visualize a graph in the 3D space regardless of the output device (i.e. an LCD monitor or a virtual reality headset such as Oculus Rift). Of course, in the case of a virtual reality headset the user will have a 3D perception of the graph thanks to stereoscopic vision. In addition, because the plug-in enables different input devices, if it detects a natural user interface (i.e. Leap Motion), the system gives visual feedback using visual icons. Whenever it recognizes one or two hands, and whether they are open or closed, two icons represent hands open or closed are placed on the right and left corners (see right and left corners of the screenshots in Figures 1 and 2). In the following, we give an accurate description of the design process used to model the interactions in the plug-in 3D Graph Explorer.

3.1. Interaction Design

3D Graph Explorer is a plug-in that allows a user to freely explore and interact with a 3D visualization of a graph obtained using the built-in Gephi layout algorithms. The graph is visualized in real time and the user can navigate around it and navigate through the network using a free-fly 3D camera providing six degrees of freedom: forwards/backwards, slide left/right, up/down (jump/crouch/lie), yaw (turn left/right), pitch (look up/down), and roll (lean left/right). This interactive method of graph exploration in the 3D space is based on the type of interaction found in computer games, where control is mostly reduced to operating standard input devices: the mouse/ keyboard and sometimes a joystick. Typical computer games that use this type of interaction are action games known as First Person Shooters. As the genre has been developed, this method of interaction has been refined and is fairly standardized.

A requirement that was identified in the design phase was the need for fast interactions, hand-eye coordination and reaction speed, which is a primary model in action games [20]. Such an approach enables a 3D human-computer

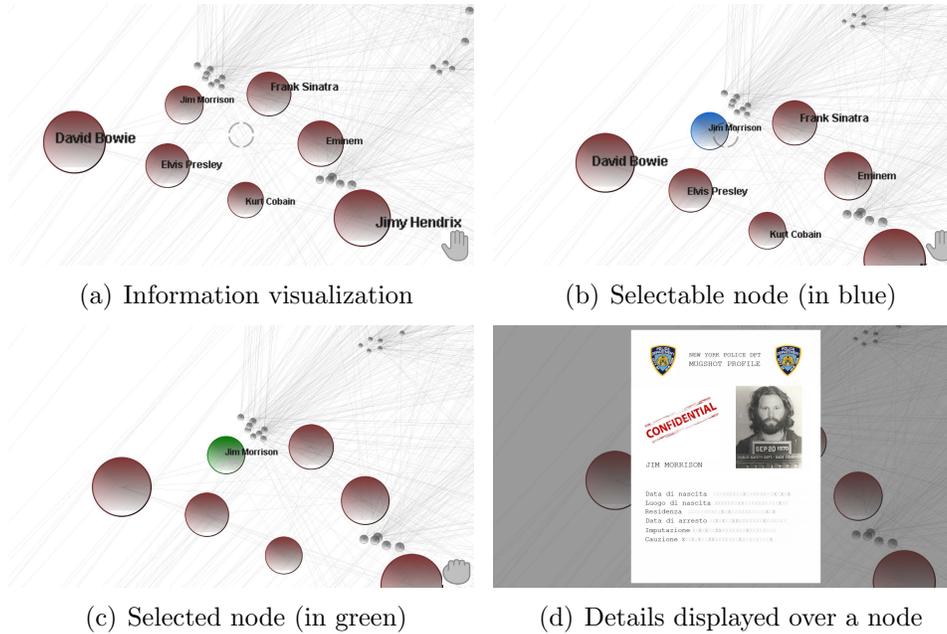


Figure 1: The visualization of a graph using the plug-in 3D Graph Explorer. Note that in this case, a natural user interface is used as input device because the system visualizes the status of the right hand in the bottom right corner. The screenshots illustrate a sequence of operations performed by the user to obtain detailed information about a node. In a), the user sees labels displayed alongside nodes. In b), the user using the open hand pointer moves the circular area as a viewfinder. In c), to select the node the hand is closed. Finally, in d) to obtain detailed information, the user points the index finger of the left hand.

interaction in which the user's tasks (e.g. moving, resizing, and grouping nodes) are performed directly in the 3D environment [21] with no need for the graphical control elements found in classical human-machine interactions. For this reason, the nodes of the graph can be selected using a resizable circular area that is used as viewfinder. Then, independently from the input device used to move camera (i.e. Leap Motion or Oculus Rift), nodes that fall within it, or that touch its border, are selectable. Once selected, nodes can be resized, moved, or clustered, or further information can be accessed (see Figures 1 and 2). However, the mouse/keyboard is not the only input device supported; in the following we discuss this traditional approach to interaction, together with natural gestures and virtual reality.

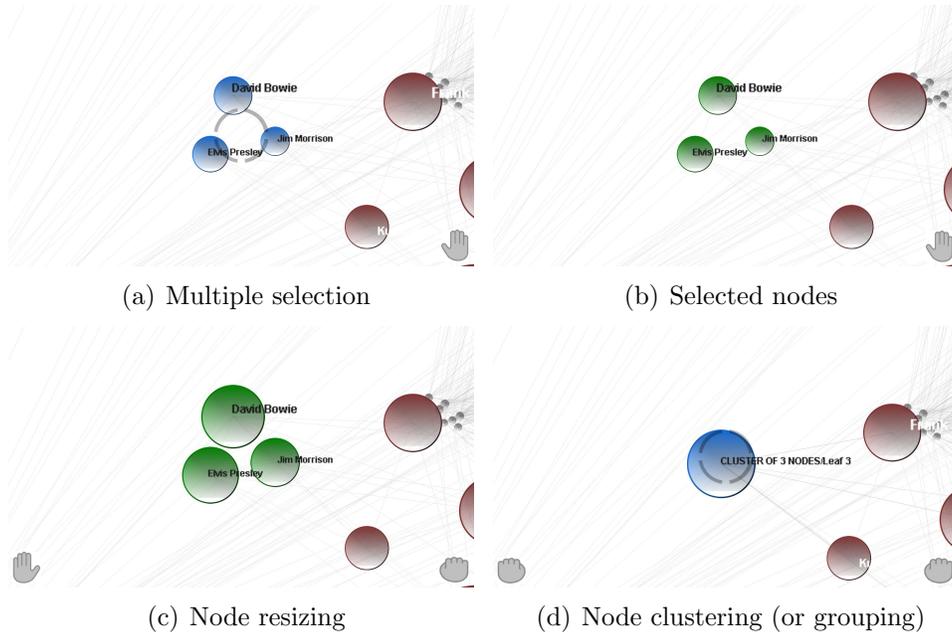
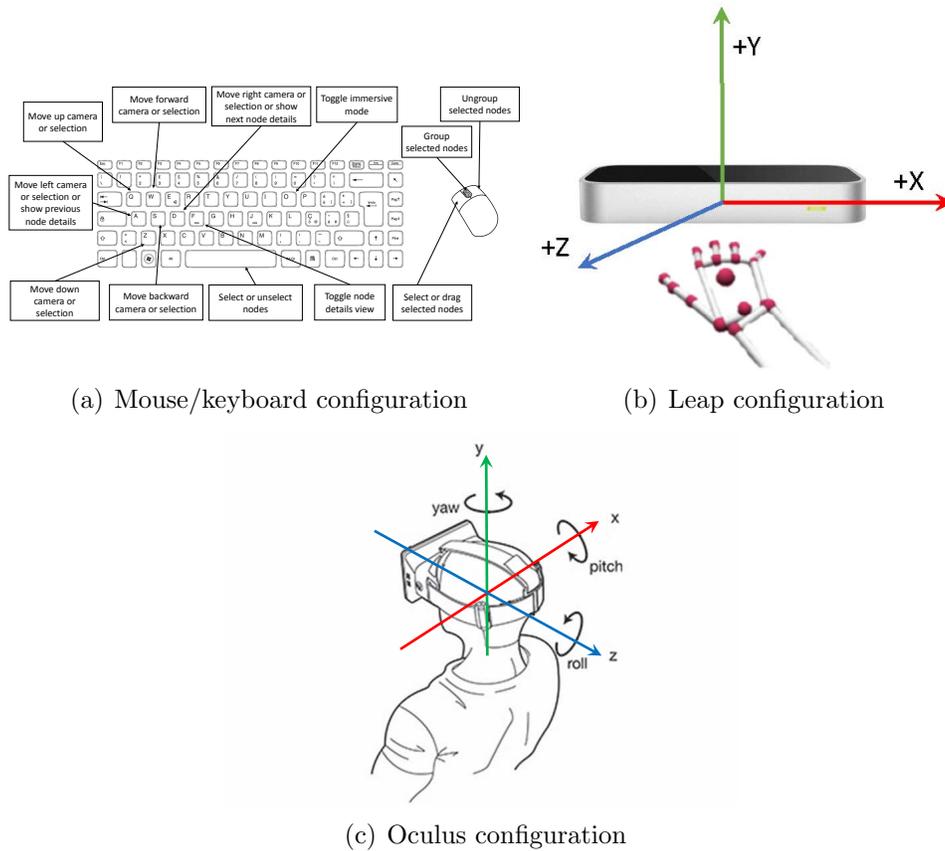


Figure 2: The screenshots illustrate resizing and clustering operations performed by the user after selected multiple nodes using a natural user interface. In a), the user using the open hand pointer moves the circular area over three nodes. Once the user closes their hand, an act of grabbing is performed and one or more nodes are selected. In b), the screenshot shows selected nodes. Resizing and clustering are performed using bimanual gestures. In the case of resizing c), the user holds the right hand and moves the left hand vertically to increase or decrease the node size as illustrated with the two icons on the bottom right and left corner. In clustering d), the left hand is closed and the same gesture is performed to cluster or decluster nodes. Clustered nodes are grouped and visualized as a single node. Clusters can be created taking others clusters as input. The declustering operation returns nodes to their original positions.



(a) Mouse/keyboard configuration

(b) Leap configuration

(c) Oculus configuration

Figure 3: The three input devices supported by the 3D Graph Explorer plug-in.

Mouse/Keyboard Interaction. This configuration makes it possible to explore a graph using a combination of keyboard and mouse (see Figure 3(a)). The keyboard keys “WASD” are used for a forward step, a side step (strafe) left, a step back, and a side step right, respectively. The user’s line of sight is controlled by the mouse and moving it rotates the user’s head (or camera). The circular area is fixed in the center of the screen and works as described previously. In addition, we used the keys Q and E to move the camera up and down, respectively. Lastly, the spacebar key is used to select/unselect nodes. Mouse buttons are used in the following way: the left button is used to drag and drop a node, the middle button is used to group nodes, and the right button is used to ungroup nodes.

Natural Gestures Interaction. Leap Motion is an USB device released in 2013 by Leap Motion Inc. The device is able to track all 10 fingers up to a precision of 1×10^{-3} of a millimeter [9]. It has a wide (150°) field of view and provides three types of spatial information: the location of fingers, the hand, and pen-like objects in Euclidean space; motion vectors for individual fingers and pen-like objects; and spherical representations of hand curvature. These features enable users to interact with their computer via hand gestures such as pinching or swiping. The device has an open API for developers and a web store offers applications ranging from 3D graphic manipulation to motion games. Interactions that are designed using Leap Motion must take into account several factors that make the user experience difficult and frustrating. Two issues are of particular concern: the typology and quantity of gestures. Typology concerns the choice of gesture selected to perform an operation. In [22], authors performed an extensive study in interacting with large datasets using gestures such as mid-air and pan-and-zoom techniques. The results suggested that bimanual interaction (with two hands) and linear gestures resulted in significantly improved performance. Another key aspect to take into account is “Gorilla Arm Syndrome” that causes strain on the arm and shoulder due to long periods of time where the hands are in front of the body in order to perform tasks [23]. Therefore the set of actions must be designed to avoid gestures that are overly complex and require high accuracy. On the other hand, quantity concerns the number of gestures. which must take into account the memory capacity of the user. [24] showed that the user can remember an average of three gestures and they suggest that this number should not be exceeded when designing an application.

Given the above, we designed a set of user-friendly gestures that allow rapid interaction with the 3D Graph Explorer plug-in. Moreover, in order to provide better support for the visual experience, the system gives visual feedback whenever it recognizes one or two hands, and whether they are open or closed (see right and left corners of Figures 1 and 2).

For graph exploration, the user can move or rotate the camera using hand gestures. In both cases, the user’s hand is open and Leap Motion tracks the palm posture, together with the accompanying roll and pitch orientations. In other words, all of the information regarding the Cartesian position and orientation of the user’s palm is retrieved from the sensor. When the user moves the camera, we use the Cartesian position in order to enable the user to move the camera backward and forward (see Figure 4(a)). As we will discuss later, when coupled with a virtual reality headset, the user is more

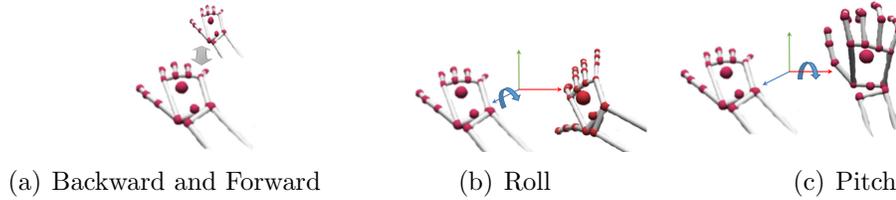


Figure 4: The hand gestures for 3D camera interaction. In (a) the user moves the camera forward and backward. In (b, c) the user rolls and pitches the camera.

comfortable with this approach. However, when the user rotates the camera, we use the Cartesian orientation in order to enable the user to roll and pitch the camera (see Figures 4(b) and 4(c)). Once the user closes their hand, the system detects an act of grabbing and one or more nodes are selected. Therefore, it switches from exploration mode to manipulation mode. The user can move the nodes they have selected in the circular area using the same operations used to move the camera. In addition, resizing, clustering and detailed information actions are allowed via a set of bimanual gestures, as illustrated in Figure 5. Finally, in order to avoid “Gorilla Arm Syndrome”, we defined the minimum arm extension in all directions and orientations. This means that an elbow-based gesture works as well as a shoulder-based gesture. Note that all of these gestures were designed to be as intuitive as possible to most users (see Section 5).

Note that despite our primitives to support graph selection and manipulation are similar to those adopted in the interaction with touch-based handsets and both techniques can thus leverage familiar interactions, the experience is essentially different. We often interact with digital objects that are predominately 2D, using a mouse or a tablet, but daily we perform many tangible 3D operations such as folding, grabbing, flipping, and stacking. Both touch and tangible input are interaction techniques that may allow similar operations for both 2D and 3D virtual objects but there are some differences. In 2D, we usually have to touch objects to move or manipulate them, for example, when folding a sheet of paper. The ability to directly touch digital information is one of the appealing characteristics of digital touch-based interaction. In a 3D virtual environment, due to the almost unconstrained freedom of interaction we have in the real world, it is easy to frustrate or confuse people through a disparity of performed action and system response. This is em-

phasized in particular by the stereoscopic vision of the head mount display that enables users to a real 3D vision of the digital objects. Then, the user has a sensation to touch an object in 3D but due to a lack of real physically feedback of action it could have a cognitive disconnect and hence worsening continuity of action. In addition to the selection/manipulation gestures, we also designed a set of gestures to enable the user to move (or to move its point of view) inside the virtual environment trying to tackle one of the biggest problems about VR: how to move around in first-person without physically walking around. Then, using the same tracking device we tried also to give, as far as we know, a new solution to this problem.

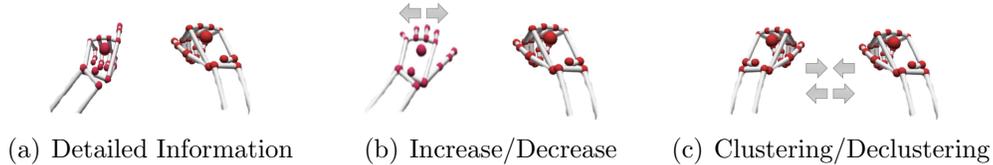


Figure 5: Bimanual gestures to manipulate the graph after nodes have been selected with a closed hand. Using the index finger of the left hand (a) detailed information can be shown. Node resizing (b) is performed by moving one hand close to the node, and moving the other vertically. Clustering (c) is performed with both hands closed and bringing them closer together.

Virtual Reality Interaction. We used the Oculus Rift virtual reality headset, released in 2012 by Oculus VR, which was developed principally for gaming. The Oculus Rift Developer’s Kit 1 (Oculus DK1) provides an extended (110°) field of view, stereoscopic vision, and responsive head tracking. The device is easy to use, inexpensive and can be calibrated with a simple tool. The kickstarter success of the Oculus Rift and the first version of the device reinvigorated interest in virtual reality experiences and paved the way for new, head-mounted displays.

The integration of the Oculus DK1 into 3D Graph Explorer takes two forms. First, the user can immerse themselves in graph exploration using stereoscopic vision. Second, it can track head movements and change the user’s point of view in a realistic way. The head-mounted display also has an input device that replaces the mouse in the mouse/ keyboard configuration. The Oculus DK1 was integrated into 3D Graph Explorer as follows: The user,

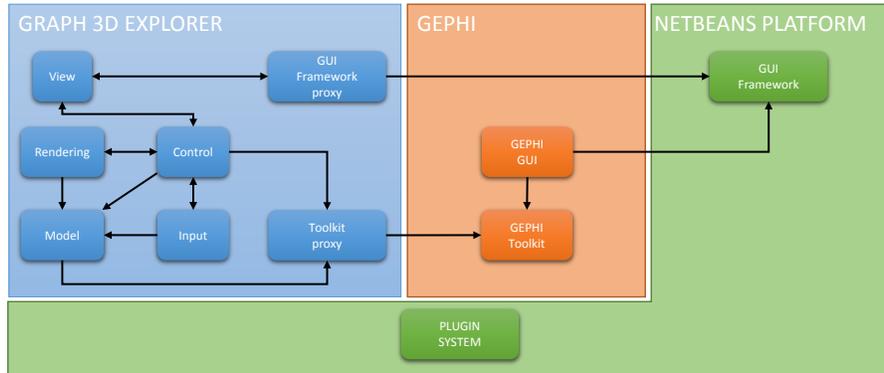


Figure 6: The 3D Graph Explorer architecture.

who is sitting in a chair switches the rendering from the LCD monitor to the virtual reality headset. The device tracks head rotation and tilts (see Figure 3(c)), which enables users to stand or sit and freely look around the virtual environment where the graph is rendered. Stereoscopic vision means that the user is totally immersed in the graph and has a better perception of its 3D representation. Because the device does not track the user’s movement, other input control devices are required such as Leap Motion. In particular, we use only to move backward and forward the camera as discussed above. In this way, the user’s head can rotate the camera while their hand can move it. This solution enables the user to navigate through the 3D space using gestures rather than mouse and keyboard shortcuts.

4. System Architecture

The architecture of the 3D Graph Explorer plug-in is shown in Figure 6. It is based on the Netbeans Platform, which is a powerful framework for developing Java-based applications [25]. The Platform provides a plug-in system and a GUI Framework for the development of user interfaces. Gephi has two distinct components: the Gephi Toolkit and the Gephi GUI. The Gephi Toolkit is a standalone library that provides data structures and algorithms for graph analysis. The Gephi GUI represents the Gephi’s user interface that interacts directly with the Netbeans Platform GUI Framework.

3D Graph Explorer is located on top of the Netbeans Platform and interacts with the Netbeans Platform GUI Framework and the Gephi Toolkit

component. 3D Graph Explorer is fully decoupled from these two components, thanks to two abstraction layers called the *GUI Framework Proxy* and the *Toolkit Proxy*. These design choices enable easy porting to other visualization systems, if needed. Internally, 3D Graph Explorer is composed of five subsystems: Model, View, Rendering, Input, and Control. We describe each in the following paragraphs.

Model. The model subsystem contains all of the logical components and abstractions needed by the plug-in. For example, it contains the 3D graph object and the state of the free-fly 3D camera that the user can move.

View. The View subsystem is the only one that interacts with the GUI Framework libraries. It manages the user interface and maintains the main window that contains the region in which the graph is displayed.

Rendering. The rendering subsystem is responsible for displaying the graph. It provides hardware-accelerated rendering based on the Java OpenGL (JOGL) graphics library, which enables the use of OpenGL with the Java language.

Input. The Input subsystem manages all of the input devices. It contains the algorithms used to perform gesture detection with the Leap Motion device and includes the logic for keyboard-mouse interaction. Furthermore, there is an abstraction of the Oculus Rift headset that controls the camera using the user's head movements.

Control. The Control subsystem coordinates all of the other components. For instance, at start-up it creates the Model and initializes the Input and Rendering components. Furthermore, when the graph currently displayed is modified externally by another Gephi plug-in, it detects these events and notifies the other subsystems.

4.1. Gephi integration

3D Graph Explorer can operate in two modes: *Gephi Panel* (see Figure 7) and *Fullscreen*. When the plug-in is in *Gephi Panel* mode, the main window depends on the main Gephi UI. The 3D Graph Explorer panel can be docked or float according to the user workflow. When the plug-in is in *Fullscreen* mode all of the toolbars disappear and the graph region fills the entire screen. Thus, this mode provides the user with a fully immersive view.

The user can switch between these two modes on the fly using the keyboard. Furthermore, if the plug-in is run on a multi-monitor configuration,

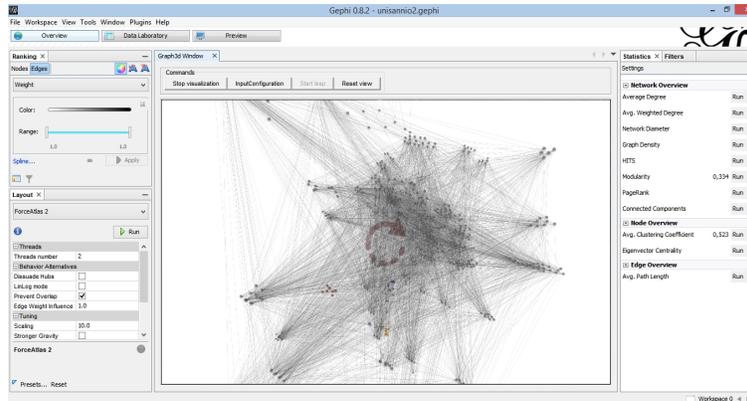


Figure 7: Gephi integration of 3D Graph Explorer.

	TRConf LCD/Mouse/Keyboard	VRConf Oculus Rift/Leap Motion
Camera forward	Key w	Push open hand forward
Camera backward	Key s	Push open hand backward
Camera orientation	Mouse tracking	VR head tracking
Select/Unselect node	Key space	Close/open hand
Clustering nodes	Central mouse button	Close hands & narrowing
Unclustering nodes	Right mouse button	Close hands & widening

Table 1: The user actions enabled in the *TRConf* and *VRConf* and how they are mapped in the corresponding input devices.

when fullscreen mode is set, the graph region is displayed on the second monitor in order to improve multitasking.

3D Graph Explorer is fully interoperable with other Gephi plug-ins. For instance, if the user chooses a graph layout algorithm, the changes to the graph are displayed in real time in 3D Graph Explorer’s main window. Furthermore, there is a bidirectional synchronization between the standard Gephi visualization tool and the 3D Graph Explorer main window. It is important to note that 3D Graph Explorer is a plug-in. It does not replace the standard Gephi visualization, but should be used in conjunction with it.

5. Evaluation

This section begins with a description of the methodology that we employed for our evaluation, while in the next section we discuss the results

obtained. We evaluated two configurations for the exploration of, and interaction with, a large amount of graph data. Specifically, we addressed the following questions:

- Are interaction modalities that are based on innovative natural user interfaces and virtual reality more difficult to use than traditional ones, that involve the use of mouse and keyboard input devices to explore a large amount of data?
- What is the reaction of users to the combined gesture-based interface and virtual reality, for the navigation of structured data in a 3D environment?
- If there is a positive evaluation, what should be done to encourage the use of these innovative interaction modalities for 3D graph exploration?

5.1. Methodology

The objective of this study was to explore the efficacy of combining natural user interfaces and virtual reality in the context of 3D data visualization. To this end we designed an experiment that compared an innovative configuration with a traditional one. The innovative configuration (*VRCConf*) exploited Leap Motion as the input device and the Oculus Rift headset as both input and output device. The traditional configuration (*TRConf*) was based on traditional devices, i.e. a keyboard, mouse, and LCD monitor. Technical details concerning interaction modalities were described in Section 3.1. The Table 1 summarizes the actions that can be performed in both configurations and that are strictly necessary to accomplish the evaluation. Note that in order to reduce the training phase we adopted only the actions necessary to complete the tasks of the evaluation.

To mitigate the impact of individual differences and to increase the output of the test results, a within-group design [26] was selected and, therefore, each participant tested both configurations.

The two configurations were tested by volunteers, and our comparison aimed to understand the differences between them. The assessment was based on constructs from the technology acceptance model (TAM) [27], which include perceived usefulness and perceived ease of use.

The TAM model is a theoretical construct that is widely used to explain and/ or predict user’s behavioral intentions when accessing a new technology or system. It has been applied in numerous studies that test user acceptance

of information technology, for example, word processors [27], spreadsheet applications, email, Web browsers, websites, e-collaboration, and blackboard. In TAM, perceived usefulness refers to “*the degree to which a person believes that using a particular system would enhance his or her job performance*”, while perceived ease of use refers to “*the degree to which a person believes that using a particular system would be free of effort*”. According to Davis [27], a system that is perceived to be easy to use is also likely to be accepted by users, and a user-friendly design can increase system acceptance. However, further additional explanatory variables may be needed. For example, Moon and Kim [28] extended TAM to a worldwide web (WWW) context, by introducing “*playfulness*” as an intrinsic belief factor that affected user acceptance. Bruner and Kumar [29] extended TAM to hand-held Internet devices and introduced a “*fun*” attribute as one of the motivational factors in their adoption.

We extended the TAM model in order to analyze whether perceptions of playfulness and attitudes to use (in addition to perceived ease of use and perceived usefulness) influenced behavioral intentions to use the test configurations (see Appendix A.3).

5.1.1. Procedure

The study was conducted in the ISISLab research laboratory at the University of Salerno, Italy. A personal computer was used, equipped with an i7, 3.40 GHz QuadCore CPU, a GeForce GTX 650 Ti graphics card, and 8.00 GB of main memory. The input devices were a standard keyboard and mouse. A Full-HD 1920x1080 LCD monitor was used for output. The test graph showed Java dependencies (*JPLD* graph), modified to be rendered in a 3D environment¹². The graph is composed of 1538 nodes and 8032 edges and was rendered at a fixed frame rate of 60 frames per second. We want to emphasize that our aim was to analyze users’ interactions regardless of the navigated graph and derive information about which interaction modality was most effective and usable.

The study consisted of three phases, namely: (a) a Preliminary Survey; (b) a Testing Phase; and (c) a Summary Survey, as defined and implemented in other contexts [30, 31, 32].

¹Java Programming Language Dependency graph (V. Batagelj)

²Available at: <http://www.di.unisa.it/~delmal/research/usability/NUI/JPLG.gexf>

In the first phase, we asked participants to fill in a preliminary questionnaire in order to collect: (a) demographic information (i.e., gender, age, education level); (b) information and communications technology (ICT) expertise; (c) general attitudes towards video games; and (d) general familiarity and experience with both virtual reality and graph theory. The 18 questions (listed in Appendix A.1) took various forms; some were open-ended, others required a “yes” or “no” answer, others required stating a preference from up to 10 possible choices, and finally, others were rated on a 5-point Likert scale with *strongly agree*/ *strongly disagree* as verbal anchors.

In the Testing Phase, users were given a 10 minute training period to become familiar with the configuration. Then they were asked to complete the following three tasks:

T1 (Search): Given the *JPLD* graph, “try to find four black nodes within a fixed amount of time, and try to read their labels”.

T2 (Exploration): Given the *JPLD* graph, “explore it in order to select as many labels as possible (mild recommendation), within a fixed amount of time”.

T3 (Clustering): Given the *JPLD* graph with 2 groups of grey nodes, “try to find and cluster them”.

The first task aimed to reveal the capacity of users to orient themselves. The second aimed to test their ability to move through a dense graph in order to have an idea of the structure of the graph. We have to emphasize that labels were not counted by users but just selected. Conversely, the system counted the number of nodes selected by the participants at the study. Moreover, labels were not selected (and therefore counted) multiple times, since after their selection from users, the system changes their color. In this way, users are informed that the nodes cannot be selected anymore.

Finally, the third task was designed to test complex user interactions. For example, the first step of this task required the selection of nodes themselves, which, in the *VRCOnf* required a two-handed interaction. In all tasks, the user was not told the maximum time allowed (5 minutes for the *Search* and *Clustering* tasks, and 2 minutes for the *Exploration* task). This allowed us to assess the degree of the user’s involvement, without creating anxiety during task execution.

At the end of each task, we asked participants to rate how easy it was to perform. Other questions addressed the responsiveness of the configuration,

how natural the interaction was, and finally, whether they experienced any problems during task execution (dizziness, nausea, tiredness, movement limitations, etc.). The first three questions were rated on a 5-point Likert scale, while the fourth offered up to 9 different choices (see Appendix A.2).

Users were monitored during the experiment and could call for assistance if they did not understand any of the instructions. Testing was performed in an isolated environment in our research lab in order to avoid distractions due to the presence of other people. Users were also encouraged to provide informal feedback (such as general comments or suggestions).

Finally, in the third phase we asked users to complete a summary questionnaire. This asked them to state their preferences from a maximum of 5 options and included three other questions rated on a 5-point Likert scale (see Appendix A.4).

The preliminary (first phase) questionnaire was only distributed to participants once. However, the questionnaires used in the second and the third phases were distributed after each test configuration. Each configuration was tested once by each user on the same day, and each test lasted between 30 and 35 minutes. The full test schedule took a week. The questionnaires that were used are reported in Appendix A.

5.1.2. Recruitment

Subjects were students of Computer Science, Mathematics, Electrical Engineering, and Chemistry at the University of Salerno, Italy. They were recruited through word of mouth advertising and students' mailing lists. Their participation was voluntary and anonymous, and participants were not compensated for taking part. They were informed that all the information they provided would remain confidential.

5.1.3. Data Analysis

Non-parametric analyses were applied to study differences between the test configurations. The Shapiro-Wilk goodness-of-fit test assessed the normality of data [33]. Regression analysis analyzed the influence of the independent variables, namely: perceived usefulness, perceived ease of use, attitude toward use, and playfulness (PU, EOU, ATT, PP), on the dependent variable, behavioral intention (BI). The internal consistency of multi-item scales was checked using Cronbach's alpha [34]. Finally, questionnaire responses

were analyzed using SPSS version 20³.

6. Results

In this Section we discuss the results from each of the three test phases.

6.1. Preliminary Survey

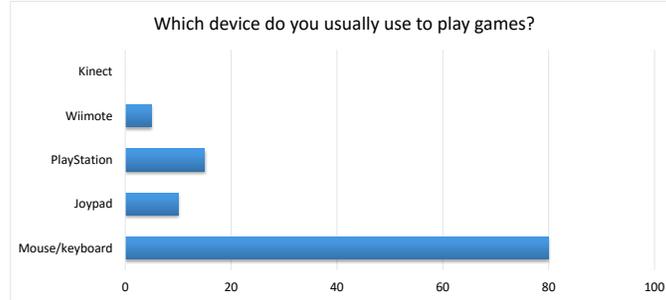
Recruits comprised 20 undergraduate (20%) and master’s level (80%) students in the Computer Science, Electrical Engineering, Chemistry, and Mathematics departments at the University of Salerno. Most of the sample was male (90%) with an average age of 24. Table 2 shows that half of respondents spent less than 7 hours per week playing video games, and that more than half considered themselves “*Competent*” in ICT.

Table 2: Participant’s Demographics

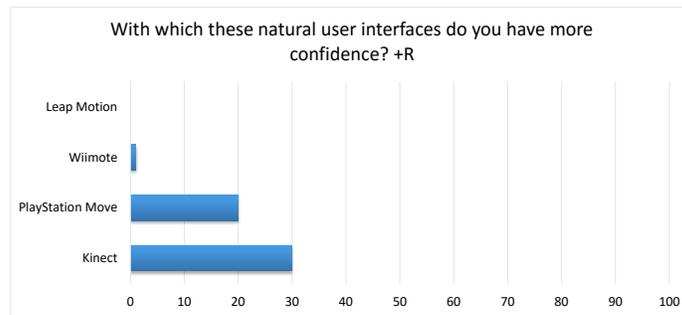
	Number	Percentage
Total Participants	20	
<i>Gender</i>		
Male	18	90%
Female	2	10%
<i>Age</i>		
20-23 years old	8	40%
24-26 years old	9	45%
26+ years old	3	15%
<i>Education Level Attained</i>		
Bachelors	4	20%
Masters	16	80%
<i>Time spent playing video games per week</i>		
0-7 hours	10	50%
8-14 hours	7	35%
14+ hours	3	15%
<i>ICT Expertise</i>		
Beginner	1	5%
Competent	13	65%
Expert	6	30%

The Preliminary Survey showed that although participants were very familiar with video games (35% rated themselves as expert in the field), they mostly used traditional input devices, i.e., mouse and keyboards (See Figure 8(a)). When asked about their familiarity with natural user interfaces, no-one was familiar with Leap Motion (see Figure 8(b)).

³<http://www-01.ibm.com/software/analytics/spss/>



(a) Devices used to play video games.



(b) Familiarity with Natural User Interfaces.

Figure 8: Preliminary Survey: general attitudes toward video gaming and familiarity with Natural User Interfaces.

With respect to their familiarity with graph theory, almost all students reported that they were very familiar (95%), while “information overload” was the most frequently-encountered problem when interacting with a graph (see Figure 9). Finally, 25% of respondents stated that they were familiar with Virtual Reality, while 90% rated themselves as inexperienced in the use of the Oculus Rift headset.

Three questions included in the Preliminary Survey (i.e., Q1, Q2, and Q3 in Appendix A.1) provided input to the k-means clustering algorithm [35]. As a result, we identified a group of *Gamers* (55%) and a group that we called *Casual Gamers* (45%). We have to emphasize that the whole sample was familiar with video games, therefore, this analysis aimed to identify the percentage of heavy users of computer games. The ANOVA analysis,

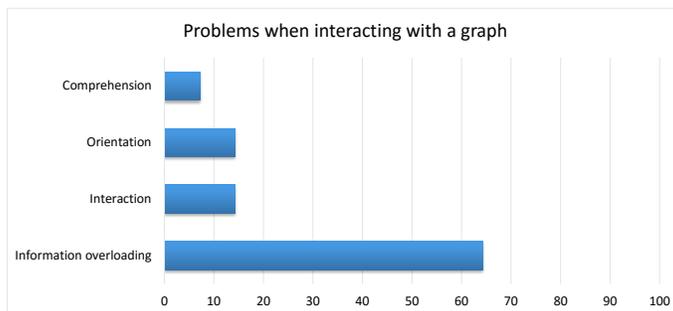


Figure 9: Problems when interacting with a graph.

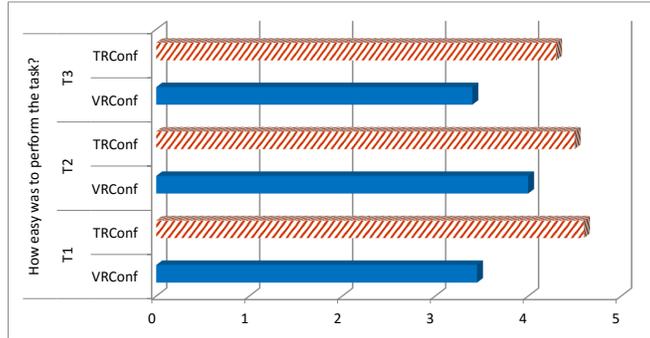
obtained with the application of the k-means clustering, showed that the variable that contributed most to the identification of clusters was question Q3, which asked participants how many hours per week they spent playing video games. Some interesting results regarding these two subsamples are presented later in this section.

In summary, the first phase allowed us to build a profile of our participants. Specifically, we identified a sample of serious video gamers, who had strong technical skills, and were very familiar with graph theory (not surprising given the high percentage of Computer Science students). Additionally, most participants used traditional devices to play games, while familiarity and experience with the Oculus Rift headset was low.

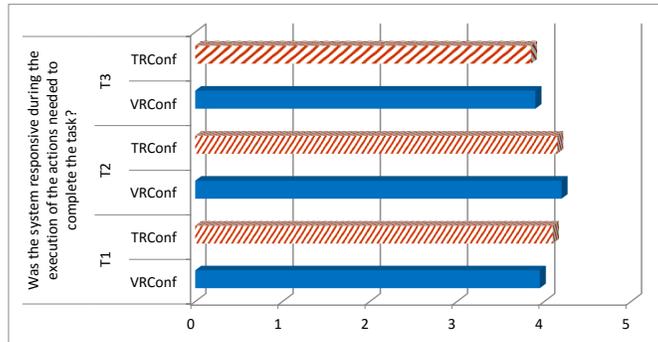
6.2. Testing Phase

The second phase involved interactions with the new system. Participants were asked to perform three different tasks (see Appendix A.2), and to evaluate: easiness (Figure 10(a)); the responsiveness of the overall configuration (Figure 10(b)); and how natural interactions with the configuration were (Figure 10(c)).

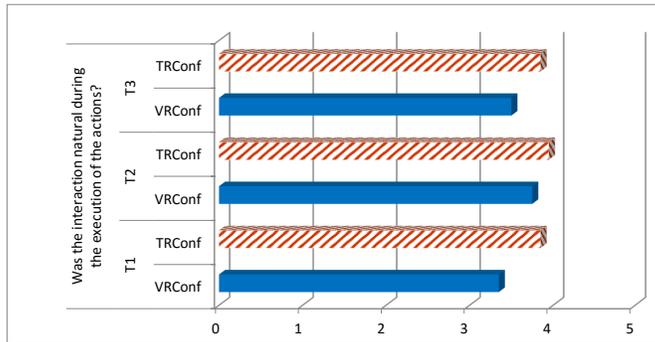
Results showed that both groups rated all of the questions positively (Figure 10). We analyzed whether there were any statistically significant differences between the groups. This showed there were no statistical differences with regard to questions about the responsiveness of the configuration, or the naturalness of movement when interacting with the data. Both groups expressed more or less the same opinions about these metrics. With regard to the question about the naturalness of the interaction, we have to emphasize that, although participants that tested *TRConf* rated slightly better



(a) Easiness of the tasks.



(b) Responsiveness.



(c) Natural interaction.

Figure 10: Comparison of test configurations on a 5-point Likert scale.

this question, participants testing *VRConf* provided positive ratings too. For the traditional configuration (*TRConf*) the naturalness was positively

rated since the high experience of participants with videogames (questions Q1-Q4) and with the use of traditional devices (question Q5). Additionally, the play mechanisms implemented in our system are the same as those used in First Person Shooter and Role-Playing Games. These categories of games had been played by more than half of participants. Participants that tested *VRConf*, found out natural the movements thanks to the provided gestures. Specifically, the most natural gesture - rotating the Oculus Rift headset - is achieved by moving the head, while grasping an object simply requires closing the hand.

On the other hand, a statistical difference was found with regard to how easy it was to perform the tasks ($p < .001$, $p < .02$, and $p < .005$, for T1, T2, and T3, respectively). Tasks were rated less positively by students using *VRConf*, which is not surprising given their inexperience with this type of device (this was confirmed by questions Q13, Q14, and Q15 in the Preliminary Survey).

Moreover, when comparing the groups derived with the clustering algorithm (i.e., *Gamers* against *Casual Gamers*) we found out very similar results in terms of Easiness and Naturalness of interactions (i.e., *TRConf* simpler and with more natural interactions against *TRConf*). Conversely, Responsiveness was evaluated better by the Gamers group when testing *VRConf*.

Questions related to problems that were experienced during task execution (T1_d, T2_d, and T3_d in Appendix A.2), showed that 13% of participants testing *TRConf* experienced movement limitations, mainly because they were used to a high-precision (i.e., high DPI) gaming mouse. In tests of *VRConf*, 37% of participants expressed concerns about both high levels of sensitivity and movement limitations. These problems were mainly due to the limited scope of Leap Motion, and their lack of experience with this type of interaction (Q13 in Appendix A.1).

During this phase, user's sessions were monitored in order to measure the time required to complete the task and precision. For T1, precision is measured by the number of nodes that participants were able to find, while for T3, it is given by the number of clusters they were able to build. As Figure 11 shows, users that tested *VRConf* required more time to complete tasks. In T1, it took them about 136 seconds to find the required (four black) nodes, compared to about 26 seconds for participants testing *TRConf*, while precision was the same (i.e., 4) for both configurations (see Table 3). Similar behavior was found in T3, where there was an order of magnitude difference in terms of mean completion time, also shown in Table 3, while the same

number of clusters were built. We recall that we did not measure the time for T2 since for this task we fixed a limit of two minutes to accomplish it.

For T2, while participants testing *VRCConf* were able to identify an average of 272 nodes, those testing *TRCConf* identified about 700 nodes. Finally, we found significant statistical differences between the test configurations for all tasks ($p < .001$).

We have to emphasize that we did not experience learning effects during our evaluation since, for each task and for each configuration, the graph is generated in a random way. Additionally, learning effects did not take place across tasks, since they were designed to evaluate different aspects (i.e., the capacity of users to orient themselves, the ability of users to quickly move through a dense graph, the testing of complex user interactions). Finally, the graph drawn randomly was composed of 1538 nodes and 8032 edges, making it very difficult to remember the positions of the nodes across the different performed runs.

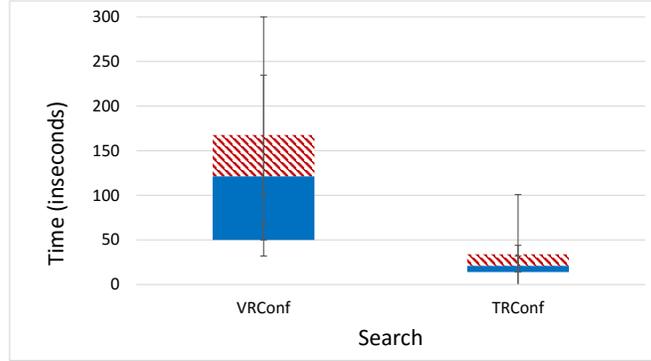
In summary, this part of the study showed that the innovative interaction modalities were found to be more difficult, and required more time to complete the tasks, while precision did not differ. The poorer performance seen in the *VRCConf* configuration may be related to several factors, including inexperience with this mode of interaction (see Fig. 8), and the distraction created by the Oculus Rift headset. In fact, although participants were encouraged to complete the task as quickly as possible, they took their time to explore the network as the experience was enjoyable.

Table 3: Task completion times and accuracy. Statistically significant difference at .001 level for all tasks.

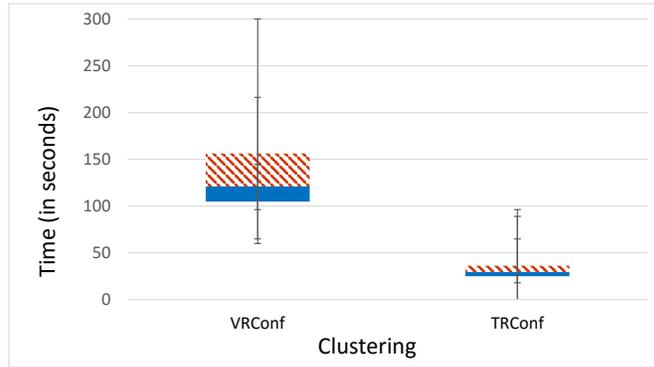
	T1 (Search)		T2 (Exploration)	T3 (Clustering)	
	Time	Precision	Number of nodes	Time	Precision
<i>VRCConf</i>	135.9	4	272	138.6	3
<i>TRCConf</i>	25.7	4	693	34.2	3

At the end of the second phase we asked participants to respond to our modified version of the TAM questionnaire. Reliability (Cronbach’s Alpha) for both groups are shown in Table 5. These are above the recommended value of 0.70 given in the literature [36]. The correlations between the various subscales for both configurations are presented in Table 4.

Intuitively, an increase in perceived usefulness and perceived ease of use would suggest an increase in the behavioral intention to use (Spearman’s



(a) Time required to complete T1.



(b) Time required to complete T3.

Figure 11: Comparison of test configurations for tasks T1 and T3 (time in seconds).

Table 4: Correlation coefficients between subscales for *VRConf* and *TRConf*. PU, Perceived Usefulness; EOU, Perceived Ease Of Use; ATT, Attitude toward use; PP, Perceived Playfulness; BI, Behavioral Intention To Use.

Subscale	VRConf					TRConf				
	PU	EOU	ATT	PP	BI	PU	EOU	ATT	PP	BI
PU	1.0					1.0				
EOU	0.512	1.0				0.596	1.0			
ATT	N.S.	N.S.	1.0			0.808	0.637	1.0		
PP	0.488	N.S.	0.565	1.0		N.S.	N.S.	0.614	1.0	
BI	0.721	0.543	N.S.	N.S.	1.0	0.897	0.506	0.795	N.S.	1.0

correlation 0.7213 $p < .02$ for *VRConf*, and 0.8970 $p < .0000001$ for *TRConf*).

As shown in Table 5, results were highly positive for all metrics. Specifi-

Table 5: Measures and constructs.

Metric		Mean (SD)		Chronbach α		Sig. Level
		<i>VRConf</i>	<i>TRConf</i>	<i>VRConf</i>	<i>TRConf</i>	
Perceived Usefulness (PU)						
PP1	Using the system would enable me to accomplish tasks more quickly	4.2 (1.7)	5.0 (1.5)	0.95	0.95	N.S.
PP2	Using the system would improve my performance	4.2 (1.6)	5.1 (1.7)			
PP3	Using the system would enhance my effectiveness	4.2 (1.4)	4.8 (1.7)			
PP4	Using the system would make it easier to do my work	4.7 (1.7)	4.9 (1.7)			
PP5	I would find the system useful in my work/experience	4.5 (1.4)	4.7 (1.4)			
Perceived Ease of Use (EOU)						
EOU1	Learning to operate the system would be easy for me	5.5 (1.6)	6.6 (0.5)	0.90	0.92	<.01
EOU2	I would find it easy to get the system to do what I want it to do	4.3 (1.4)	6.3 (0.8)			<.001
EOU3	My interaction with the system would be clear and understandable	5.4 (1.4)	6.5 (0.7)			N.S.
EOU4	I would find the system flexible to interact with	4.9 (1.6)	6.2 (1.0)			<.005
EOU5	It would be easy for me to become skilful at using the system	6.2 (1.3)	6.7 (0.6)			N.S.
EOU6	I would find the system easy to use	5.5 (1.4)	6.6 (0.6)			<.01
Perceived Playfulness (PP)						
PP1	When interacting with the system, I do not realize the <i>time that has elapsed</i>	6.0 (1.4)	5.5 (1.5)	0.81	0.80	N.S.
PP2	When interacting with the system, I am not aware of <i>any noise</i>	6.1 (1.4)	5.9 (1.5)			N.S.
PP3	Using the system makes my task <i>fun</i>	6.7 (0.6)	5.7 (1.7)			<.005
PP4	Using the system stimulates my <i>curiosity</i>	6.5 (1.4)	5.6 (1.5)			<.01
PP5	Using the system leads to <i>exploration</i>	6.1 (1.4)	5.2 (1.6)			<.02
PP6	Using the system stimulates my <i>imagination</i>	5.9 (1.7)	4.8 (1.6)			<.005
Attitude Toward Using (ATT)						
ATT1	Using the system is a (bad/good) idea	5.9 (1.1)	5.5 (1.3)	0.76	0.93	N.S.
ATT2	Using the system is a (foolish/wise) idea	5.5 (0.8)	5.3 (1.0)			
ATT3	Using the system is a (unpleasant/pleasant) idea	5.9 (1.1)	5.5 (1.2)			
ATT4	Using the system is a (negative/positive) idea	6.3 (1.0)	5.8 (1.1)			
Behavioral Intention to Use (BI)						
B1	I will use the system on a regular basis in the future	5.2 (1.4)	5.1 (1.6)	0.86	0.92	N.S.
B2	I will strongly recommend others to use the system	5.5 (1.2)	5.2 (1.7)			

cally, perceived usefulness (PU) and perceived ease of use (EOU) were rated more positively in the traditional configuration, while playfulness (PP), attitude to use (ATT) and (to a slight extent) behavioral intention to use (BI) were rated more highly in the Virtual Reality configuration. In the *VRConf*, the most positively rated questions were, *perceived playfulness*, PP3 (*Using the system makes my task fun*), PP4 (*Using the system stimulates my curiosity*), *Attitude Toward Using*, and ATT4 (*Using the system is a (negative/positive) idea*).

Figure 12 shows the average metrics for both configurations; this Figure provides a snapshot of the differences between the two configurations. It shows that we did not find any statistical differences between the two groups with regard to the perceived usefulness (PU) of the configuration and the behavioral intention (BI) to use it. Conversely we did find a statistical difference with regard to perceived ease of use (EOU) and perceived playfulness (PP). These results confirm earlier observations, i.e. participants testing *VRConf*

took extra time to complete the task as the curiosity, fun and novelty of the idea led them to explore the network, without realizing how much time had elapsed.

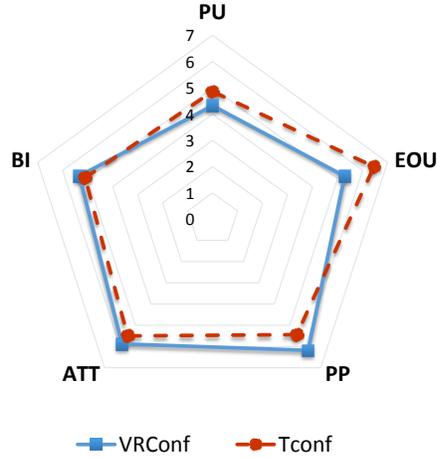


Figure 12: TAM results across all subscales, 7-point scale.

A regression analysis was carried out in order to identify which variables influenced the use of a specific configuration. In this case, the dependent variable was the behavioral intention (BI) metric. Independent predictor variables were the experience of gaming (i.e., *Gamers* or *Casual Gamers*), and the TAM subscales (i.e., perceived usefulness, perceived ease of use, attitude to use, and perceived playfulness). The regression analysis in Table 6 shows that the best predictor for BI was perceived usefulness in both configurations. The model yielded an adjusted R^2 value of 0.592 for *VRConf* and 0.769 for *TRConf*.

Therefore, our analysis shows that perceived usefulness is a significant variable in both configurations. This is consistent with several other empirical studies which have demonstrated that perceived usefulness, but not perceived ease of use, is positively related to behavioral intentions to use an information system or information technology [37, 38, 39].

Taking into account the two subsamples (*Casual Gamers* and *Gamers*) we repeated the regression analysis to see if the results that were obtained for the full sample remained valid. A comparison showed that perceived usefulness had an effect on BI for *Casual Gamers* but not *Gamers*. This result is valid for both test configurations.

Table 6: Results of multiple linear regression analysis. B, Unstandardized coefficient; β Standardized coefficient; SE, Standard Error; Adjusted $R^2 = 59.2\%$ for *VRConf* and 76.9% for *TRConf*.

Predictor variables	VRConf					TRConf				
	B	SE(B)	β	<i>t</i>	<i>P</i>	B	SE(B)	β	<i>t</i>	<i>P</i>
(Constant)	.522	1.456		3.59	.725	-1.269	2.256		-.563	.583
Video gamer	.430	.424	.180	1.013	.328	-.102	.405	-.031	-.251	.805
Usefulness	.567	.204	.679	2.778	.015	.784	.195	.690	4.030	.001
Ease Of Use	.011	.259	.011	.043	.967	.057	.377	.020	.151	.882
Attitude	.431	.381	.272	1.130	.277	.424	.356	.272	1.192	.253
Playfulness	-.078	.295	-.062	-.265	.795	-.018	.248	-.012	-.072	.943

In summary, the important message derived with this analysis is that by intervening on perceived usefulness and perceived ease of use metrics (given their positive correlation), individuals that are not familiar with gaming would be more likely to use the proposed system than gamers, despite their inexperience.

6.3. Summary Survey

In this section, we report the results of the questionnaire submitted in the third phase. In general, as shown in the previous section, all participants positively rated the perceived usefulness and the perceived ease of use of the test configurations. Statistical differences between configurations for all questions are shown in Table 7.

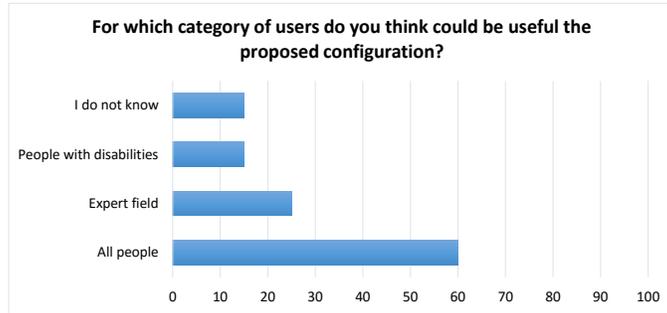
Table 7: Summary Survey Results

Question	VRConf	TRConf	Sig. Level
Do you think it was useful to use the proposed system?	4.0	4.1	N.S.
Do you think it was interesting to use the proposed system?	4.1	3.5	<.05
Do you think it was easy to use the proposed system?	3.2	4.1	<.05

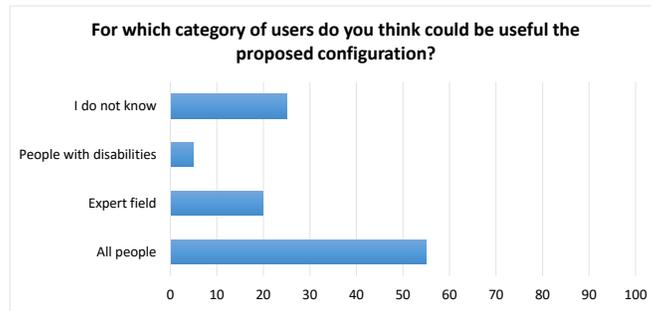
Additionally, as shown in Figure 13, when asked the question: “*Which category of users do you think the proposed system could be useful for?*”, more than half of participants stated that the test configurations could be useful for all users, not just domain experts.

7. Conclusion and Future Works

In this paper, we presented a novel approach to real-time 3D exploration and interaction of large graphs using a natural user interface and immersive



(a) Opinion of users testing *VRCOnf*.



(b) Opinion of users testing *TRCOnf*.

Figure 13: Who would benefit most from the proposed configuration?

virtual reality. The approach was implemented as a plug-in, named 3D Graph Explorer, for Gephi, an open-source graph and network analysis software package. Our approach was tested using an empirical study based on a comparison of different configurations for the exploration of, and interaction with a large graph. Specifically, we designed an innovative configuration (*VRCOnf*), which used Leap Motion as the input device and the Oculus Rift headset as both input and output devices, and compared it to a traditional configuration (*TRCOnf*), based on the use of a keyboard, mouse, and LCD monitor.

Our evaluation showed that participants' who tested *VRCOnf* found the use of innovative technologies more challenging than traditional devices, while performing the same tasks. While *TRCOnf* was assessed as more useful and easy to use, the Oculus Rift headset and the Leap Motion configuration was found to be more fun, and stimulated curiosity, exploration, and imagina-

tion (confirmed by answers to open-ended questions). Participant’s responses were positive with respect to both configurations, and they reported the same willingness to tell others about them and use them in the future. We showed that with improvements to perceived usefulness and perceived ease of use (the factors that were rated less positively), we are able to encourage their use, mainly by people with less experience in the field. Specifically, users would be more inclined to use intuitive interfaces designed for interactions based on simple gestures. The high potential of gesture-based interfaces and the playfulness of the virtual reality environment made the exploration of a large amount of data an enjoyable experience for all users. Such a system could be used in other contexts, such as the simulation of autonomous characters [40], which would offer benefits to all users, not only experts in the field of 3D graph exploration.

We have to emphasize that this work has some limitations. First, all participants at our preliminary study were students from an Italian academic environment. Our samples were composed of users with high education levels and with an age ranging from 20 to 30 years. Moreover, they were mostly gamers, ICT skilled, and very familiar with graph theory. Therefore, our results may not necessarily be representative of the entire world population. Future work could look into attitudes and behaviors about using natural user interfaces with regard to older age groups (i.e., general audience), students from other academic areas (not only from a scientific field) and with even more diversified technological skills. A long-term experiment, with a longer and accurate training phase, will be designed and performed accordingly. Finally, a larger number of subjects would provide more statistically significant results. Moreover, in addition to widening the testing pool, another interesting direction could be to widen the set of innovative interfaces in our study, in order to produce a thorough analysis about which interface is most preferred by final users for 3D graph exploration.

As future works, we are going to perform experiments in order to test how certain 3D graph layout algorithms help or hinder users to fulfill certain graph reading tasks through empirical human experimentation with a virtual reality headset. Finally, we could also to widen the set of graphs, in order to derive insights on which type of graph (sparse, dense, and so on) could benefit of 3D exploration.

References

- [1] K.-L. Ma, C. W. Muelder, Large-scale graph visualization and analytics, *Computer* 46 (7) (2013) 39–46.
- [2] G. Di Battista, P. Eades, R. Tamassia, I. G. Tollis, Algorithms for drawing graphs: An annotated bibliography, *Comput. Geom. Theory Appl.* 4 (5) (1994) 235–282.
- [3] M. Bastian, S. Heymann, M. Jacomy, Gephi: An open source software for exploring and manipulating networks (2009).
URL <http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154>
- [4] A. Garg, R. Tamassia, Graph Drawing: Symposium on Graph Drawing, GD '96 Berkeley, California, USA, September 18–20, 1996 Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, Ch. GIOTTO3D: A system for visualizing hierarchical structures in 3D, pp. 193–200.
- [5] M. Patrignani, F. Vargiu, Graph Drawing: 5th International Symposium, GD '97 Rome, Italy, September 18–20, 1997 Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, Ch. 3DCube: A tool for three dimensional graph drawing, pp. 284–290.
- [6] I. Bruß, A. Frick, Fast interactive 3-d graph visualization, in: Proceedings of the Symposium on Graph Drawing, GD '95, Springer-Verlag, London, UK, UK, 1996, pp. 99–110.
- [7] P. Luckey, Oculus Rift (2012).
URL http://en.wikipedia.org/wiki/Oculus_Rift
- [8] Microsoft Kinect (2010).
URL <http://www.microsoft.com/en-us/kinectforwindows/>
- [9] M. Buckwald, D. Holz, The Leap Motion 3D Controller (2010).
URL <https://www.leapmotion.com/>
- [10] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, T. Ideker, Cytoscape: a software environment for integrated models of biomolecular interaction networks, *Genome research* 13 (11) (2003) 2498–2504.

- [11] D. Auber, D. Archambault, R. Bourqui, A. Lambert, M. Mathiaut, P. Mary, M. Delest, J. Dubois, G. Mélançon, et al., The tulip 3 framework: A scalable software library for information visualization applications based on relational data.
- [12] M. Chimani, C. Gutwenger, M. Jünger, G. W. Klau, K. Klein, P. Mutzel, The Open Graph Drawing Framework (OGDF), CRC Press, 2013.
- [13] J. Raskin, The Humane Interface: New Directions for Designing Interactive Systems, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 2000.
- [14] M. Nancel, J. Wagner, E. Pietriga, O. Chapuis, W. Mackay, Mid-air pan-and-zoom on wall-sized displays, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2011, pp. 177–186.
- [15] L. Gallo, A. P. Placitelli, M. Ciampi, Controller-free exploration of medical image data: Experiencing the kinect, in: Computer-Based Medical Systems (CBMS), 2011 24th International Symposium on, IEEE, 2011, pp. 1–6.
- [16] G. C. S. Ruppert, L. O. Reis, P. H. J. Amorim, T. F. de Moraes, J. V. L. da Silva, Touchless gesture user interface for interactive image visualization in urological surgery, World journal of urology 30 (5) (2012) 687–691.
- [17] A. C. de Carvalho Correia, L. C. de Miranda, H. Hornung, Gesture-based interaction in domotic environments: State of the art and hci framework inspired by the diversity, in: Human-Computer Interaction–INTERACT 2013, Springer, 2013, pp. 300–317.
- [18] D. Bassily, C. Georgoulas, J. Guettler, T. Linner, T. Bock, Intuitive and adaptive robotic arm manipulation using the leap motion controller, in: ISR/Robotik 2014; 41st International Symposium on Robotics; Proceedings of, VDE, 2014, pp. 1–7.
- [19] B. Shneiderman, The eyes have it: a task by data type taxonomy for information visualizations, in: Visual Languages, 1996. Proceedings., IEEE Symposium on, 1996, pp. 336–343.

- [20] A. Rollings, E. Adams, Andrew Rollings and Ernest Adams on Game Design, New Riders Games, 2003.
- [21] D. A. Bowman, S. Coquillart, B. Froehlich, M. Hirose, Y. Kitamura, K. Kiyokawa, W. Stuerzlinger, 3d user interfaces: New directions and perspectives, *IEEE Comput. Graph. Appl.* 28 (6) (2008) 20–36.
- [22] M. Nancel, J. Wagner, E. Pietriga, O. Chapuis, W. Mackay, Mid-air pan-and-zoom on wall-sized displays, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '11*, ACM, New York, NY, USA, 2011, pp. 177–186.
- [23] M. F. Shiratuddin, K. W. Wong, *Transactions on edutainment viii*, Springer-Verlag, Berlin, Heidelberg, 2012, Ch. Game Design Considerations when Using Non-touch Based Natural User Interface, pp. 35–45.
- [24] J.-F. Jego, A. Paljic, P. Fuchs, User-defined gestural interaction: A study on gesture memorization, in: *3D User Interfaces (3DUI)*, 2013 IEEE Symposium on, 2013, pp. 7–10.
- [25] H. Bck, *The Definitive Guide to NetBeans Platform 7*, 1st Edition, Apress, Berkely, CA, USA, 2011.
- [26] J. Lazar, J. H. Feng, H. Hochheiser, *Research methods in human-computer interaction*, John Wiley & Sons, 2010.
- [27] F. D. Davis, Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology, *MIS Q.* 13 (3) (1989) 319–340.
- [28] J.-W. Moon, Y.-G. Kim, Extending the {TAM} for a World-Wide-Web context, *Information and Management* 38 (4) (2001) 217 – 230.
- [29] G. C. B. II, A. Kumar, Explaining consumer acceptance of handheld Internet devices, *Journal of Business Research* 58 (5) (200) 553 – 558.
- [30] D. Malandrino, V. Scarano, R. Spinelli, How Increased Awareness Can Impact Attitudes and Behaviors Toward Online Privacy Protection, in: *Proceedings of the 2013 International Conference on Social Computing, SOCIALCOM'13*, Washington, DC, USA, 2013, pp. 57–62.

- [31] D. Malandrino, M. Ilaria, G. Palmieri, V. Scarano, G. Filatrella, How Quiz-based Tools can improve students' engagement and participation in the classroom, in: 2014 International Conference on Collaboration Technologies and Systems (CTS), 2014, pp. 379–386.
- [32] A. Fish, C. Gargiulo, D. Malandrino, D. Pirozzi, V. Scarano, Visual Exploration System in an Industrial Context, *IEEE Trans. Industrial Informatics* 12 (2) (2016) 567–575.
- [33] S. S. Shapiro, M. B. Wilk, An Analysis of Variance Test for Normality (Complete Samples), *Biometrika* 52 (3/4) (1965) pp. 591–611.
- [34] L. J. Cronbach, Coefficient alpha and the internal structure of tests, *Psychometrika* 16 (3) (1951) 297–334.
- [35] L. G. Berry, M., *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management*, Wiley, 1997.
- [36] I. H. B. Jum C. Nunnally, *Psychometric theory*, McGraw-Hill, 1994.
- [37] P. J. Hu, P. Y. Chau, O. R. L. Sheng, K. Y. Tam, Examining the technology acceptance model using physician acceptance of telemedicine technology, *Journal of management information systems* (1999) 91–112.
- [38] P. Y. Chau, An empirical assessment of a modified technology acceptance model, *Journal of management information systems* (1996) 185–204.
- [39] P. Y. Chau, P. J.-H. Hu, Investigating healthcare professionals decisions to accept telemedicine technology: an empirical test of competing theories, *Information & management* 39 (4) (2002) 297–311.
- [40] U. Erra, B. Frola, V. Scarano, BehaveRT: A GPU-based Library for Autonomous Characters, in: *Proceedings of the Third International Conference on Motion in Games, MIG'10*, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 194–205.

Appendix A. Surveys

Questions of questionnaires have been translated from the Italian language.

Appendix A.1. Preliminary Survey Questionnaire

- Q1: Do you like videogaming?
Strongly disagree Strongly agree
- Q2: How do you consider your experience with videogames?
Inexpert Expert
- Q3: How many hours per week do you spend by playing videogames?
 Less than one hour Between one and seven hours
Between eight and fourteen hours
 Between fifteen and twenty one hours More than twenty one hours
- Q4: Which type of videogames do you play?
 First-Person-Shooter Adventure Role-Playing Games
Strategy/Tactics Sports
 Fighting Dance/Rhythm Survival Horror
Other (Add here which one)
 None (I do not play any videogame)
- Q5: Which type of device do you use when playing videogames?
 Mouse/keyboard Kinect Joypad PlayStation Other
(Add here which one) None
- Q6: Are you left-handed?
Yes No
- Q7: Do you have vision deficiencies?
– *Same answer options as Q6*
- Q8: If you answered “Yes” to the previous question, which type of deficiency do you have?
– Open Question
- Q9: Do you wear glasses?
– *Same answer options as Q6*
- Q10: How do you consider your IT experience?
– *Same answer options as Q2*
- Q11: Have you heard of the term *graph*?

- *Same answer options as Q6*
- Q12: How do you consider your experience with graph theory?
 - *Same answer options as Q2*
- Q13: With which these natural user interfaces do you have more confidence?
 - Kinect Leap Motion PlayStation Move Other (Add here which one)
 - None (No familiarity with natural user interfaces)
- Q14: Do you have familiarity with *virtual reality*?
 - *Same answer options as Q1*
- Q15 Have you heard of the Oculus Rift?
 - *Same answer options as Q6*
- Q16: If you answered “Yes” to the previous question, which is your experience with the Oculus Rift?
 - *Same answer options as Q2*
- Q17: Have you ever had need to interact and visualize data on a graph?
 - *Same answer options as Q6*
- Q18: If you answered “Yes” to the previous question, which type of problems do you encountered?
 - Interaction problems (In terms of personal difficulties) Comprehension
 - Information overloading
 - Few useful information Orientation
 - Other (Add here which one) None

Appendix A.2. Task activities Questionnaire

Task One. Given the *JPLD* graph, “try to find four black nodes within a fixed amount of time, and get close to them in order to visualize their labels.”

- T1.a: How easy was to perform the task?
Very difficult ○ ○ ○ ○ ○ Very easy
- T1.b: Was the system responsive during the execution of the actions needed to complete the task?
Strongly disagree ○ ○ ○ ○ ○ Strongly agree
- T1.c: Was the interaction natural during the execution of the actions?
Strongly disagree ○ ○ ○ ○ ○ Strongly agree

- T1_d: Have you experimented one of the following contraindications? Please select which one(s).
 - High sensibility Lack of the field of view of the device Movements limitations Motor difficulties Dizziness Nausea Tiredness Other None

Task Two. Given the *JPLD* graph, “*explore it in order to visualize as many labels as possible, within a fixed amount of time.*”

- T2_a: How easy was to perform the task?
 - *Same answer options as T1_a*
- T2_b: Was the system responsive during the execution of the actions needed to complete the task?
 - *Same answer options as T1_b*
- T2_c: Was the interaction natural during the execution of the actions?
 - *Same answer options as T1_c*
- T2_d: Have you experimented one of the following contraindications? Please select which one(s).
 - *Same answer options as T1_d*

Task Three. Given the *JPLD* graph with 2 groups of grey nodes, “*try to find and cluster them.*”

- T3_a: How easy was to perform the task?
 - *Same answer options as T1_a*
- T3_b: Was the system responsive during the execution of the actions needed to complete the task?
 - *Same answer options as T1_b*
- T3_c: Was the interaction natural during the execution of the actions?
 - *Same answer options as T3_c*
- T3_d: Have you experimented one of the following contraindications? Please select which one(s).
 - *Same answer options as T1_d*

Appendix A.3. Analyzing the acceptance of the tested configurations

Based on:

- Davis, F. D. (1989), “Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology”. *MIS Quarterly*, 13:3, 319-340.
- Ji-Won Moon, Young-Gul Kim (2000). “Extending the TAM for a World-Wide-Web context”. *Information & Management* 38 (2001) 217-230.

Perceived Usefulness (PU)

- PU1. Using the system would enable me to accomplish tasks more quickly
unlikely likely
- PU2. Using the system would improve my performance
unlikely likely
- PU3. Using the system would enhance my effectiveness
unlikely likely
- PU4. Using the system would make it easier to do my work
unlikely likely
- PU5. I would find the system useful in my work/experience
unlikely likely

Perceived Easy of Use (EOU)

- EOU1. Learning to operate the system would be easy for me
unlikely likely
- EOU2. I would find it easy to get the system to do what I want it to do
unlikely likely
- EOU3. My interaction with the system would be clear and understandable
unlikely likely
- EOU4. I would find the system to be flexible to interact with
unlikely likely
- EOU5. It would be easy for me to become skillful at using the system
unlikely likely
- EOU6. I would find the system easy to use
unlikely likely

Perceived Playfulness (PP)

- PP1. When interacting with the system, I do not realize the *time elapsed*
unlikely likely
- PP2. When interacting with the system, I am not aware of *any noise*
unlikely likely
- PP3. Using the system gives *fun* to me for my task
unlikely likely
- PP4. Using the system stimulates my *curiosity*
unlikely likely
- PP5. Using the system leads to my *exploration*
unlikely likely
- PP6. Using the system arouses my *imagination*
unlikely likely

Attitude Toward Using (ATT)

- ATT1. Using the system is a (bad/good) idea:
bad good
- ATT2. Using the system is a (foolish/wise) idea:
foolish wise
- ATT3. Using the system is a (unpleasant/pleasant) idea:
unpleasant pleasant
- ATT4. Using the system is a (negative/positive) idea:
negative positive

Behavioral Intentions to use (BI)

- BI1. I will use the system on a regular basis in the future
unlikely likely
- BI2. I will strongly recommend other to use the system
unlikely likely

Appendix A.4. Post Survey Questionnaire

- P1: For which category of users do you think could be useful the proposed system?
 All people People expert of the field People with disabilities Other I do not know

- P2: Do you think it was useful to use the proposed system?
– *Same answer options as Q1*

- P3: Do you think it was interesting the proposed system?
– *Same answer options as Q1*

- P4: Do you think it was ease to use the proposed system?
– *Same answer options as Q1*