GAZE-BASED FOCUS ADAPTION IN AN INFORMATION VISUALIZATION SYSTEM

Marc Streit
Graz University of Technology
streit@icg.tugraz.at

Alexander Lex
Graz University of Technology
lex@icg.tugraz.at

Heimo Müller
Medical University of Graz
mueller@mac.com

Dieter Schmalstieg
Graz University of Technology
schmalstieg@icg.tugraz.at

ABSTRACT

As the complexity and amount of real world data continuously grows, modern visualization systems are changing. Traditional information visualization techniques are often not sufficient to allow an in-depth visual data exploration process. Multiple view systems combined with linking & brushing are only one building block of a successful InfoVis system. In this paper we propose the incorporation of cheap and simple gaze-based interaction. We employ the tracking information not for selecting data (i.e. mouse interaction) but for an intelligent adaption of 2D and 3D visualizations. Derived from the focus+context paradigm, we call this *gaze-focus*. The proposed methods are demonstrated by means of three different visualizations.

KEYWORDS

Information Visualization, Gaze Tracking, Multiple Views.

1. INTRODUCTION

Information Visualization (InfoVis) strives to visualize abstract data in an easily understandable way. Over the years, several paradigms such as linking & brushing, focus+context and details on demand were established and are widely accepted as critical success factors for an InfoVis system. Focus+context and details on demand are usually implemented based on a selection, triggered by a mouse or keyboard interaction. A user clicks some element, which is then put into focus, for example by zooming, while other elements are compacted and possibly abstracted. When analyzing this workflow, we noticed that in order to select an element one has to first identify it in the contextual representation of the data. The human vision has a rather small angle of focused sight, compared to the wider peripheral vision. We therefore implemented a system where the currently focused part in the visualization correlates as closely as possible to the fixation of our eyes, the gaze-focus. We do this by tracking the gaze of a subject and putting the element a subject is looking at, in the focus of the visualization.

2. RELATED WORK

Gaze-based interaction is not an everyday user interaction technique, primarily because of the high cost of the required systems. However, this is currently changing, as eye tracking becomes technically more feasible with low cost equipment (Hiley, et al., 2006; Kumar, et al., 2007).

Previous work (Fono, et al., 2005) has shown that gaze-based interaction with windows is preferable over traditional input techniques. Fono, et al., (2005) use gaze tracking to select and zoom windows a user is focusing on and also to zoom in a digital media application. They showed that task completion time was faster and preferred by users with gaze tracking compared to traditional input devices. However, gaze-based systems suffer from some inherent deficiencies. One problem is the involuntary selection of items (the Midas

Touch Effect), which can be overcome to some degree by, for example, selecting only after a fixation has lasted for a certain time (dwell time) or after a manual click (Vertegaal, et al., 2008). The accuracy of high quality eye tracking systems nowadays is limited to about 2 degrees (e.g. www.tobii.com); while low cost systems have a precision of about 1cm on screen (Hiley, et al., 2006). Ashmore, et al., (2005) try to overcome this by magnifying the region of the gaze with a fisheye lens and then selecting the target within the magnification. Additionally to these approaches, which use 2D distortion techniques, one of our techniques relies on an interaction with a 3D arrangement, relying on perspective effects rather than fisheye distortion.

3. METHODS

In the following we differentiate between gaze-based methods within a single view and interaction that aims for the management and handling of multiple, linked visualizations in a 3D setup.

3.1 Single view gaze interaction

Parallel coordinates are well suited to visualize several thousands of data points simultaneously over a limited number of dimensions. As the number of dimensions increases, details are lost due to the reduced spacing of axes. While truly large amounts of dimensions need special approaches, which produce a reduced subset, e.g. (Yang, et al., 2003), manual distortion can increase the number of dimensions perceivable simultaneously. In this process a large number of axes (around 40) are shown at a time. The spacing, and thereby the readability of interesting regions can be increased (creating a fisheye distortion). These properties lend themselves perfectly to gaze-based scene manipulation. Since only a small region is observed sharply in the fixation phases of the eye, this region is enlarged, once a user looks at it. The spacing of the other axes is reduced, thereby using less screen real estate while still providing the contextual information. Fig. 1 shows a screenshot of the implementation, where the gaze was close to the green vertical line.

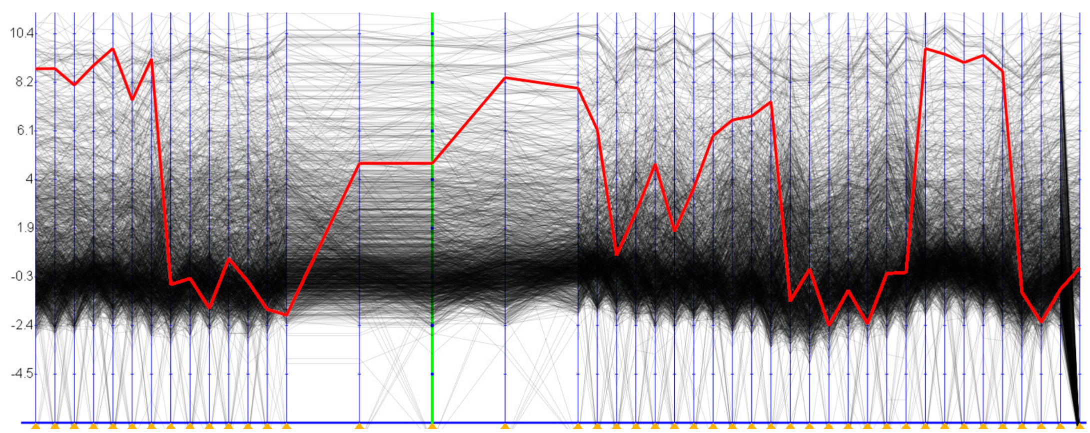

Fig. 1. 2D scene manipulation in a parallel coordinates view.
The gaze direction of the user extends the space between a predefined number of axes.

In a **heat map** the magnitude of a value is mapped to a color. Affiliation with a property is spatially encoded. As a consequence, the number of simultaneously visualized elements has a strict upper limit: the number of available pixels on the screen. In some tasks this is not sufficient. Therefore, we use a hierarchical approach with three levels, shown in Fig 2.

An overview of all 30.000 elements is rendered on the left, a data subset with up to 1000 elements in the center, and a detailed view on the right. We again employ a gaze following automatic focusing feature, to maximize the clarity of the focused visualization. When a user looks at the second level, it is enlarged, facilitating easier browsing in the dataset (Fig. 2 a). Once the user gazes at the detail view, this view is focused, thereby putting an emphasis on the individual elements (Fig. 2 b).

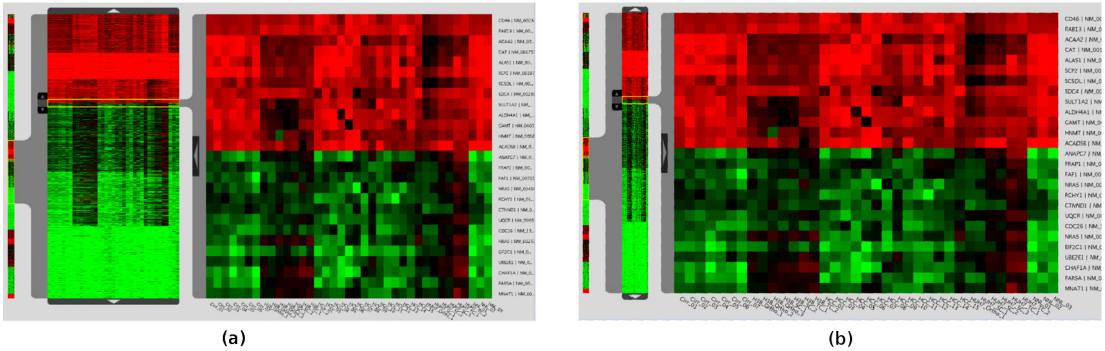

Fig. 2. 2D scene manipulation in a hierarchical heat map. (a) When looking on the overview level on the left, it is enlarged. (b) When the gaze is directed onto the detailed representation, this is enlarged in turn.

3. 2 Multiple view gaze interaction

In order to investigate different aspects of a dataset, the information is depicted in multiple visualizations. In combination with linking & brushing techniques, the user has a powerful tool to perform visual data exploration and analysis. State-of-the-art multiple view systems arrange views side by side on the screen. Fono, et al., (2005) showed how to zoom in on an application window - which can also be applied to multiple view applications. However, this method is naturally limited by the available screen space and can therefore only be used for a very low number of views. In previous work in our group we tried to overcome this problem by introducing a novel approach to manage up to about 20 related views. Planes containing the content of 2D visualizations are arranged in a bucket-like layout (see Fig. 3). The bottom of the bucket contains the view in focus. Four contextual views are forming the bucket walls. The rim of the bucket contains a thumbnail list of views that are currently not of immediate interest, but can be swapped in later on. The user can arbitrarily rearrange views inside the bucket and its rim by using drag and drop.

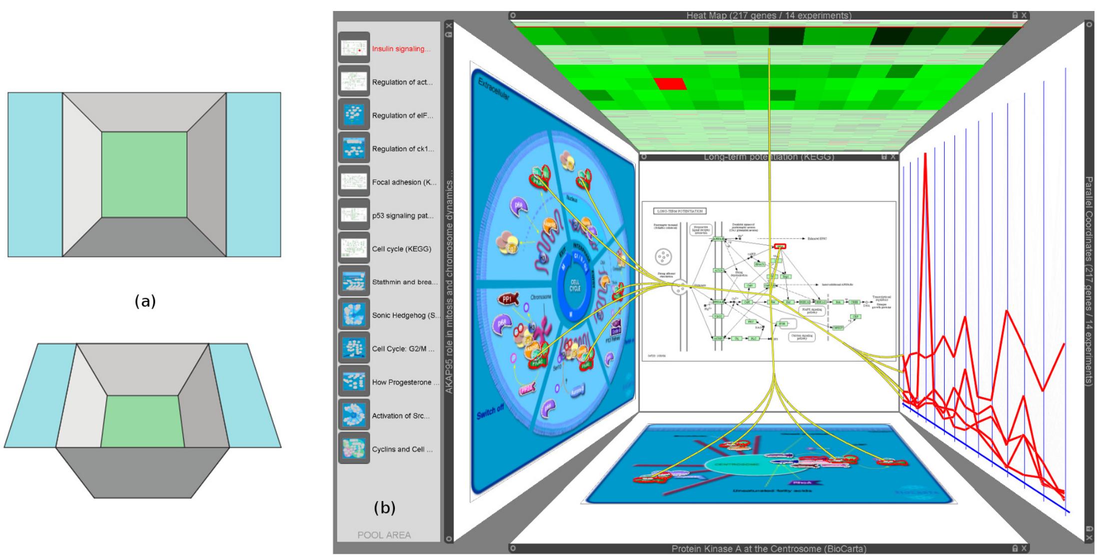

Fig. 3 (a). Bucket setup. The center of the bucket shows the view in focus (green). On the walls (gray) connected views are rendered. The rim (blue) presents a list of contextual views may be of later interest. (b) shows the bucket with real data loaded. A selected entity is highlighted in all views and connected by visual links (yellow tree).

Selected entities are synchronized between all views and highlighted accordingly. Recent research in the field of information visualization uses visual links (i.e. simple connection lines) among dependent views (Collins & Carpendale 2007). The bucket employs visual linking to emphasize relations between elements (see Fig. 3 b).

While the multi-level approach (bucket bottom, walls and rim) enables the management of numerous views, it introduces a distortion problem. Especially text is difficult to read when rendered on the bucket walls. Therefore, we extended the static bucket setup to a “rubber” bucket by taking into account the user’s gaze information. The bucket is rotated according to the gaze direction (see Fig. 4), reducing the distortion of the view looked at. When the user moves the head towards the screen, the focused visualization in the 3D scene is transformed to the user’s direction (diving into the bucket). Additionally the gaze navigation immerses the user into the scene.

4. IMPLEMENTATION

We implemented the *gaze-focus* with a very low cost hardware setup, sufficient for a proof-of-concept prototype. The gaze direction was determined by a head-mounted Nintendo® Wii™ Controller and an infrared bar mounted on top of the screen.

The presented visualizations are created in the Caleydo Visualization Framework (www.caleydo.org) (Streit, et al., 2008). Caleydo is a multiple view system supporting modern InfoVis paradigms. Although it is a general purpose visualization framework, the predominant use case deals with biological data. All figures show real world data from our life science partners.

For 3D rendering we use the Java OpenGL library (JOGL). JOGL is capable of rendering in an applet that can be run in a browser. Alternatively, the application can be delivered to the user via the JNLP webstart technology.

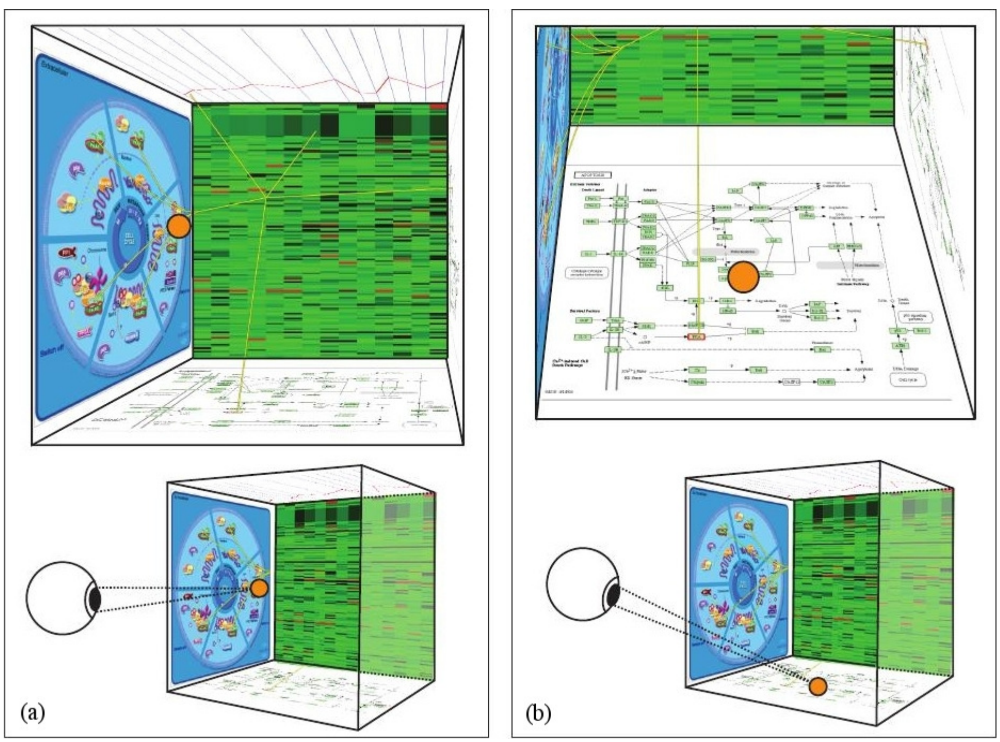

Fig. 4. The bucket is adopting according to the user’s focus point (orange). In (a) the user gazes at the left bucket plane while in (b) the user looks at the lower plane.

5. CONCLUSION

In this paper we present methods to incorporate gaze-based manipulation of 2D and 3D views in an information visualization system. While our current setup is sufficient for development, a real life system has to be less obtrusive and more precise. However, for the *gaze-focus*, we do not require the accuracy of professional eye tracking systems. Instead we aim to integrate a low cost webcam based eye tracking module in the visualization framework, thus bringing this type of user interaction to a broad public.

Although not yet formally evaluated, the positive initial feedback from our users encourages us to continue work in that direction.

ACKNOWLEDGEMENT

The authors want to thank Bernhard Kainz for his help with an early version of the hardware setup and Bernhard Schlegl for his implementation of the hierarchical heat map visualization.

This work was funded by the FIT-IT program (813 398), the Fonds zur Förderung der wissenschaftlichen Forschung (FWF) (L427-N15), and the Zukunftsfonds Steiermark (3007).

REFERENCES

Ashmore, M., et al., 2005. Efficient eye pointing with a fisheye lens. *GI '05: Proceedings of Graphics Interface 2005*. Victoria, British Columbia, pp. 203-210.

Collins, C. and Carpendale, S., 2007. VisLink: Revealing relationships amongst visualizations. *Transactions on Visualizations and Computer Graphics,* Vol. 13, No. 6.

Fono, D. & Vertegaal, R., 2005. EyeWindows: evaluation of eye-controlled zooming windows for focus selection. *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*. Portland, Oregon, USA, pp. 151-160.

Hiley, J., et al., 2006. A Low Cost Human Computer Interface based on Eye Tracking Engineering in Medicine and Biology Society. *IEEE Engineering in Medicine and Biology Society Conference*, pp. 3226-3229.

Kumar, M. & Winograd, T., 2007. GUIDe: gaze-enhanced UI design, *ACM CHI '07 extended abstracts on Human factors in computing systems*. San Jose, CA, USA, pp. 1977-1982.

Streit, M., et al., 2008. Navigation and Exploration of Interconnected Pathways, *In Computer Graphics Forum*, Vol. 27, No. 3, pp. 951-958.

Vertegaal, R., 2008. A Fitts Law comparison of eye tracking and manual input in the selection of visual targets. *IMCI '08: Proceedings of the 10th international conference on Multimodal interfaces*. Chania, Crete, Greece, pp. 241-248.

Yang, J., et al., 2002. InterRing: an interactive tool for visually navigating and manipulating hierarchical structures. *INFOVIS '02: Proceedings of the IEEE Symposium on Information Visualization*. pp. 77-84