A SOLUTION FOR THE FOCUS AND CONTEXT PROBLEM IN INTERACTIVE GEOVISUALIZATION APPLICATIONS

T. Ropinski[†], K. Hinrichs[†], F. Steinicke[†]

[†]Institut für Informatik, Westfälische Wilhelms-Universitt Münster, Germany

KEY WORDS: Three-dimensional Visualization, Graphics, Urban Planning.

ABSTRACT:

One major aim of exploring virtual geo-environments is to reveal information which is hidden in large sets of high-dimensional data. Since only three dimensions in space - plus one dimension in time - can be used in 3D visualization it is difficult to visualize a large number of attributes associated with geo-objects without burdening the user's cognition. The 3D Magic Lens metaphor has the potential to solve this problem, since it can be used to assign different visual appearances to different parts of a virtual environment. Based on an image-based magic lens rendering algorithm we revealed several application areas in geovisualization which can potentially benefit from the use of 3D magic lenses. In this paper we will give application examples of different lenses applied to virtual geo-environments. In particular, we will introduce the use of the magic lens metaphor to aid landscaping, urban city planning as well as subsurface visualization tasks. Based on these application areas we will also introduce a classification of different lens types.

1 INTRODUCTION

Because of the advances in data acquisition and data management, the amount of data which needs to be visualized by today's visualization systems is growing. Furthermore the number of attributes associated with a dataset often exceeds the dimensionality of space and time. These huge high-dimensional datasets pose a challenge for interactive visualization, because special visualization techniques are required to support the user during exploration tasks. This problem is often solved by visualizing only a subset of the attributes, which meets the demands of the user. A common technique used within visualization systems to display several of such subsets of high-dimensional datasets is the *multiple view system* (North and Schneiderman (1997)). A graphics application is a multiple view system if it supports the investigation of a given conceptual entity by two or more *distinct views*. In a multiple view system each view can be used to visualize a corresponding subset of the overall dataset. Thus distinct views help the user to learn about different aspects of the conceptual entity, which is ensured by presenting different information or emphasizing different aspects of the same information in distinct views (Wang Baldonado et al. (2000)).

However, when implementing visualization systems, where multiple views are considered, several design goals have to be taken into account. To meet design goals, one must consider two conceptual questions, asking *whether* multiple views should be used, and if so, *how* they should be used. Thus initially it has to be determined whether it is useful to integrate more than one distinct view into a particular visualization system. Although it has been discovered that multiple views improve user performance, discovery of unforeseen relationships as well as unification of the desktop, their use is not always expedient since multiple views may result in a cognitive overload for the user (North and Schneiderman (1997)). If the developer has decided to integrate multiple views, the design of these views has to be defined, i.e., it has to be determined what information is displayed in which view. Furthermore problems arise when interactions between distinct views of a multiple view system have to be defined. On the one hand these interactions are performed on a different level of abstraction, i.e., the overall graphics system in contrast to the visualizations presented inside each view, leading to a cognitive overload. On the other hand, interactive exploration techniques integrated into multiple view systems pose a challenge to the application developer. To achieve interactivity, besides adequate interaction techniques hardware resources have to be taken into account to support real-time visualization, which is necessary to give immediate visual feedback to the user. Especially when numerous views are integrated into a multiple view system interactivity can not always be guaranteed because the number of views directly determines the rendering performance.

When using multiple view systems, the same subset of attributes is visualized everywhere in the virtual environment presented in one view, although it is often desirable to visualize different sets of attributes in different parts of the scene. For example in landscaping or city planning applications it is important to visualize specific attributes in the region of interest while retaining the overall impression by using a realistic rendering for the rest of the scene to improve visual comprehension. These demands can be met by the 3D magic lens metaphor which has the potential to assign different visual appearances to different parts of one virtual environment. Thus only one view is used within the application, and the aforementioned problems can be solved by using this focus and context visualization metaphor, i.e., different subsets of potentially high dimensional datasets can be visualized inside resp. outside the lens.

In particular, interactive visualization applications can benefit from the use of the magic lens metaphor. Although many visualization techniques have the potential to enhance user cognition significantly when applied to virtual environments, 3D interaction does not necessarily benefit from the usage of the same visual appearances, because often the demands to enhance user cognition on the one hand and the demands to improve 3D interaction on the other hand diverge. Hence in many cases it is desirable to apply different visual appearances in distinct regions of a virtual environment to enhance 3D interaction as well as user cognition. By using magic lenses, it is possible to highlight special regions of interest by using a different visual appearance, while still preserving contextual information for the rest of the scene. Thereby it is possible to incorporate the visualization demands of user cognition and 3D interaction in a virtual environment to support the investigation of a single conceptual entity.

The paper proceeds in the next section by discussing related work.

Section 3 gives a brief classification of 3D magic lenses applicable in virtual geo-environments. Application areas in the field of geovisualization are described in Section 4, in particular some examples are given demonstrating how virtual landscaping, virtual city planning as well as subsurface visualization applications can benefit from the 3D magic lens metaphor. The paper concludes in Section 5.

2 RELATED WORK

As described in the preceding section, the magic lens metaphor has the potential to improve visualization systems without complicating their design process or their usage. This section discusses published work about the magic lens metaphor.

The magic lens metaphor has been introduced in 1993 by Bier et al. (Bier et al. (1993)). In their work they describe *Toolglass*™ widgets as new interface tools that can appear, as though on a transparent sheet of glass, between an application and a traditional cursor. Toolglass widgets can be positioned with one hand while the other positions the cursor. They may incorporate visual filters, known as magic lenses, which modify the visual appearance of application objects, enhance data of interest or suppress distracting information in the region of interest, which is determined by the shape of the lens. In their work Bier et al. describe how to combine toolglass widgets and the magic lens metaphor to improve human-computer interaction. When using toolglass widgets with magic lenses, two handed interaction is appropriate to perform the necessary interaction tasks. The non-dominant hand can be used to position the widget, while the dominant hand can be used to control the cursor of the mouse. As one application for two handed interaction in conjunction with magic lenses, Bier et al. introduce *click-through buttons*. These transparent toolglass widgets can be positioned interactively on top of the application data by using the non-dominant hand. To trigger one of the click-through buttons, which are arranged in a cluster like manner, the user clicks on it by controlling the mouse with the dominant hand. When such a button is used, the underlying application data is modified in a way specified by the button, e.g., the application data can be colored using a predefined color associated with the button. In addition to the possibilities given by these interaction metaphors, Bier et al. concentrate on the visual filters, which can be used as 2D magic lenses to modify the visual appearance of underlying graphics objects. In particular they describe preview lenses, which allow the user to preview changes he wants to apply to graphics objects. These lenses visualize the changes in a small region only to help the user to decide whether to apply them or not. Furthermore shadow lenses, achromatic lenses and local scaling lenses are introduced. While the former apply a shadow and change the color of the underlying graphics objects, a local scaling lens can be used to minimize or magnify underlying graphics objects or parts of graphics objects. Besides the use of magic lenses in 2D graphics applications Bier et al. describe how to apply the concept of magic lenses and toolglasses for text editing. They also give an example how to apply the concept of 2D planar lenses to 3D virtual environments. In contrast to the approaches described in this section, the magic lens is not integrated into the virtual environment but appears as a separate layer on top of the final image.

Inspired by the initial idea of the magic lens metaphor introduced in Bier et al. (1993) several papers followed, covering a taxonomy (Bier et al. (1995)) and the composition of 2D magic lenses (Fox (1998)) as well as several applications of the concept (Stone et al. (1994), Bier et al. (1997), Fuhrmann and Gröller (1998), Stoev et al. (2002)).

Cignoni et al. (Cignoni et al. (1994)) were the first to transfer the magic lens metaphor to volumetric lenses. In their work they describe the *MagicSphere* metaphor representing an insight tool for 3D data visualization. As the name implies, the metaphor is limited to a spherical lens volume. Besides the restriction to only one usable lens shape the visual appearance lacks due to their analytical approach: In a preprocessing step geometric primitives are classified based on their position relative to the border of the magicsphere. Subsequent rendering requires two passes; one for the geometric primitives lying outside the magicsphere and one for those inside the magicsphere. The elements lying on the border, classified as border elements, are rendered in both of the two passes. When using the magicsphere with a *MultiRes* filter Cignoni et al. obtain a satisfactory visual appearance even for the border elements which are rendered twice. However, the visual appearances used in the two rendering passes of their *EdgesEmphasizer* filter differ too much and therefore cause visual artifacts near the border of the magicsphere. Similar artifacts have to be expected with other visual appearances used by the magicsphere metaphor.

Another more general extension of the magic lens metaphor to 3D virtual environments has been presented by Viega et al. (Viega et al. (1996)). They introduced an algorithm for visualizing volumetric lenses as well as flat lenses in a 3D virtual environment. Their implementation of the magic lens metaphor exploits SGI Reality Engine hardware support for clipping planes. Since only infinite clipping planes are supported, it takes an extra rendering pass for almost every face of the lens volume. Thus it would be computationally very expensive to render magic lenses having arbitrary shapes. Another disadvantage of this approach results from the limited number of clipping planes supported by current graphics devices increasing the number of necessary rendering passes. Because of these reasons, Viega et al. provide only examples of box-shaped 3D magic lenses.

A different analytical approach of a similar concept has been used by Idelix Software Inc. (IDELIX (2002)). Their Pliable Display Technology 3D (PDT3D) avoids object occlusions in 3D virtual environments by analyzing camera and lens parameters and applying corresponding geometric transformations to occluding objects, as it has been proposed in (Carpendale et al. (1997)). Thus it is possible to select a region of interest to which the system provides an occlusion-free view. As mentioned above, the major disadvantage of this concept is the modification of the scene structure lying outside the region of interest through geometric transformations, which leads to a loss of contextual information.

Programming demos of concepts similar to the magic lens metaphor can be found in current graphics SDKs. Microsoft delivers their DirectX 9 SDK with a demo called *clipvolume*. It demonstrates the use of pixel and vertex shaders to eliminate certain parts of the scene from rendering. Because the decision whether a fragment should be excluded from rendering is based on its distance to the center of a spherical volume, the approach is very similar to the concept of the magicsphere metaphor and is only capable of clipping scenes against spherical volumes. The *cull fragment* demo, which comes with *n*Vidia's *n*VSDK also demonstrates the exclusion of geometry lying inside a spherical volume from rendering.

Fröhlich et al. (Fröhlich et al. (1999)) have described the application of volumetric clipping lenses to seismic datasets. In their work, they show how the exploration of seismic datasets can benefit from the use of volumetric lenses. However, their approach supports only lenses having cuboid geometries. Because the surfaces of a cube are given by orthogonal planes, the visualization is similar to the common slicing approach where multiple clipping planes are used to reveal inside structures Curtis et al. (1986). Therefore the technique introduced by Fröhlich et al. can be considered as a generalization of this slicing approach.

All the techniques described above either constrain the shape of the lens or are not sufficient to allow interactive frame rates, which are important to give the user immediate visual feedback when interacting with a virtual geo-environment. Inspired by the existing techniques for rendering 3D magic lenses, we have developed an image-based algorithm which allows to render arbitrarily shaped convex 3D magic lenses in real-time, i.e., with interactive frame rates (Ropinski and Hinrichs (2004)). Later on we have extended the algorithm to also allow rendering of arbitrary non-convex lens shapes at interactive frame rates. Based on this algorithm we have revealed several application areas in virtual geo-environments.

3 AN OVERVIEW OF VARIOUS LENS TYPES

In order to support an intuitive application of 3D magic lenses it must be easy to assign different visual appearances to a lens. Therefore we have implemented an object-oriented interface allowing the easy integration of 3D magic lenses into the 3D graphics system VRS[1]. The main capabilities of a 3D magic lens can be controlled using the methods of the class `MagicLens`. Since in this section we are only interested in classification aspects of lenses corresponding to the visual appearance assigned to it, a short overview of types of visual appearances is given. The methods of the class `MagicLens` provided for managing the visual appearance associated with a 3D magic lens, which must be applied when rendering the parts of the scene intersecting the lens volume, can be categorized into three groups corresponding to the three main types of visual appearances that can be associated with a 3D magic lens:

- *General appearances* are applied to all shapes which intersect the lens shape. General appearances are stored in the `appearance_` attribute which is of type `AttributeComposite`. They can be accessed using the methods `addAppearanceAttr`, `removeAppearanceAttr` and `getAppearance`.
- *Shape appearances* are only applied to those parts of special shapes or groups of shapes influenced by the lens. To apply a shape appearance it can be associated with a shape by providing the shape's object address or name, which can be used to identify objects in VRS, to the magic lens. The shape appearances assigned to a lens are stored in the dictionary `shapeAppearance_` and can be accessed by the methods `addShapeAppearanceAttr`, `removeShapeAppearanceAttr` and `getShapeApperance`.
- The exclude list *invalid shapes* can be considered as a visual appearance as well, since parts of the shapes stored in this list are not rendered if intersected by the magic lens. This exclude list is stored in the private array `invalidShapes_` and can be modified by calling the methods `addInvalidShape` and `removeInvalidShape`.

Based on the assigned visual appearances as well as their relation to the camera a classification is possible. Thus, 3D magic lenses can be classified into two main types: *camera lenses* and *scene lenses*. Camera lenses are positioned relative to the virtual camera, whereas scene lenses can be positioned anywhere in a virtual environment by using multimodal interactions. Camera lenses can assist the user while exploring dense datasets, e.g., seismic datasets, by translucently rendering or removing objects occluding the view of the camera. Thus it is possible to navigate through virtual environments containing dense data without removing context information. Three combinable visual appearances bound to camera lenses have been implemented: semi-transparent rendering, removing of partial or complete data intersecting the lens volume and highlighting data inside the lens volume by additional light sources. In contrast to camera lenses, scene lenses are particularly suitable for emphasizing specific regions of interest interactively.

[1] http://www.vrs3d.org/

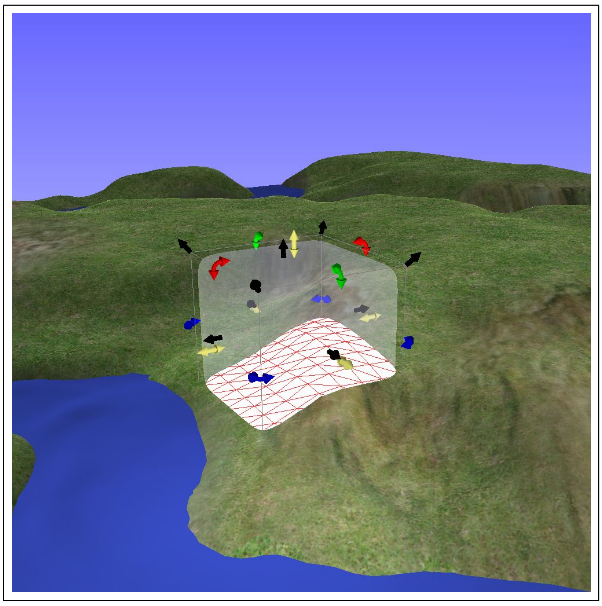
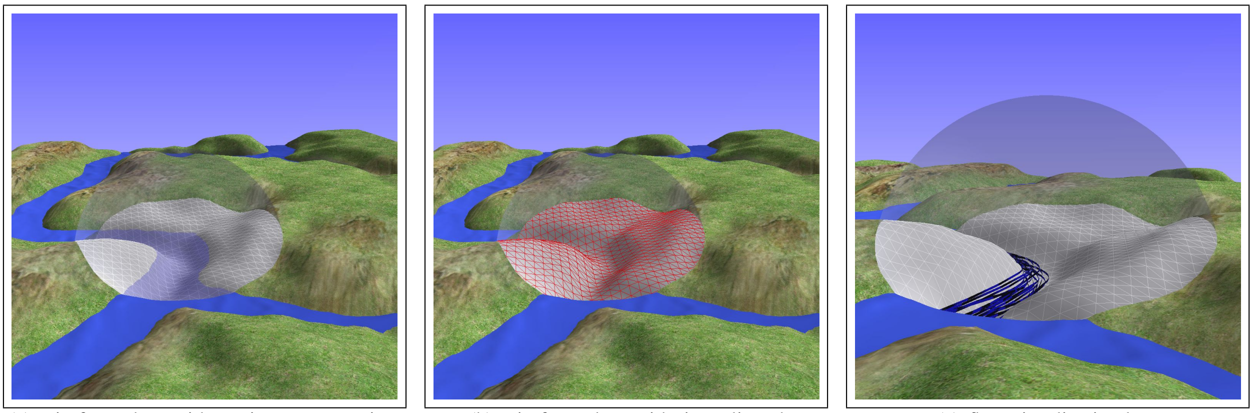


Figure 1: Interactive manipulation of lens shapes using 3D widgets.

The scene lenses described in this paper can be either positioned by using the keyboard or by using multimodal 3D widgets, which are analogous to the Toolglass™ widgets presented by Bier et al. (Bier et al. (1993)) used to position 2D magic lenses. Figure 1 shows a quadric shaped lens which can be manipulated by using the shown 3D widget. The shown widget has several handles having different colors, which can be used to perform a certain interaction. The yellow handles centered at the sides of the bounding box's faces can be used to translate the lens along the respective axis. Black handles can be used to change the size of the magic lens by applying a scaling transformation, whereas the handles on the corners of the bounding box allow a uniform scaling, and the handles positioned at the faces allow a non-uniform scaling. To rotate the lens volume the blue, red and green handles can be used to rotate around the respective axis. Since the visual representation of the widget needs a lot of display space and may distract the user's attention the widget can be turned off in case it is not needed.

For both camera lenses as well as scene lenses many different visual appearances are possible:

- wireframe lenses are capable of displaying vertex details for parts of the scene without modifying the global view, which would lead to a loss of context information,
- eraser lenses can reveal occluded parts of the scene by re-



(a) wireframe lens with semi-transparent river (b) wireframe lens with river clipped (c) flow visualization lens

Figure 2: Application of magic lens to a digital terrain.

moving partial or complete information from regions intersecting the lens,

- texture lenses can assist in exploring terrain data by replacing or modifying the texture of objects. These 3D lenses are a generalization of the 2D texture lenses introduced in Döllner et al. (2000).

Beyond the application of 3D magic lenses to modify the visual appearance of the intersecting geometry, several other application scenarios are possible. For example, *time lenses* may be used to show different states of a process, e.g., plant growth. Applications exploiting the potential of modifiable lens shapes may assist the user during interactive exploration. For example a modifiable cylinder shaped lens may assist when analyzing subsurface data, e.g., by elongating or bending the lens.

Beyond these examples, the concept of volumetric magic lenses has already been used to show different levels of detail for a model (Cignoni et al. (1994)) and to enhance flow visualization (Fuhrmann and Gröller (1998)).

4 MAGIC LENSES IN VIRTUAL GEO-ENVIRONMENTS

The powerful concept of magic lenses has been deployed in many application areas. But due to the computationally expensive realization of volumetric magic lenses, leading to a limitation of lens shapes and an overhead in rendering time, mostly 2D magic lenses have been applied so far. Based on our 3D magic lens rendering algorithm allowing interactive frame rates when applying arbitrarily shaped 3D magic lenses we have revealed several application areas in geovisualization. This section briefly discusses a few application examples based on virtual geo-environments within which we have successfully applied the 3D magic lens metaphor to improve visual comprehension.

4.1 Virtual Landscaping

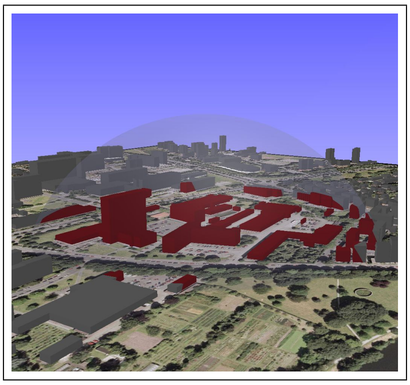
In the past years many terrain rendering algorithms have been proposed which allow to render large scale terrain models at interactive frame rates. This allows to use these models within virtual landscaping applications. In the following we will give a few examples how to combine these applications with 3D magic lenses. One of the scene lenses which has been implemented for the use in landscaping applications assists the user in exploring the structure of a riverbed. The lens can be used with different associated visual appearances that can be applied within the virtual environment. In Figure 2 four examples are shown how the 3D magic lens has been applied within an interactive landscaping application.

Figure 2 shows the landscaping application loaded with a digital terrain containing a riverbed. In Figure 2(a) the scene is shown from the same viewpoint with a spherical 3D magic lens applied. The visual appearance associated with the lens is a combination of two shape appearances: the geometry representing the water is rendered semi-transparently, and the geometry representing the terrain is rendered using an *opaque wireframe mode*. Rendering in opaque wireframe mode can be achieved by rendering the geometry twice. In the first pass the surface is rendered using a base color, in this case white. In the next pass the same geometry is rendered in wireframe mode using a different color, in this case red. This kind of rendering allows to reveal the geometric structure by showing the faces building up the geometry, without losing the depth cue given by hidden surface elimination. Thus the shown lens helps to reveal the structure of the riverbed by displaying the geometric structure of the terrain. In Figure 2(b) a similar 3D magic lens with a modified visual appearance is shown. In this case the geometry representing the river is contained in an exclude list associated with the applied lens and is therefore not rendered. Figure 2(c) shows a flow visualization lens. By using this lens the geometry approximating the surface of the river is also excluded from rendering. In contrast to the lens shown in Figure 2(b) the water is visualized as a stream model consisting of several unique streams. Using this visualization it is possible to make predictions regarding erosion within the riverbed, e.g., to plan landuse around the river.

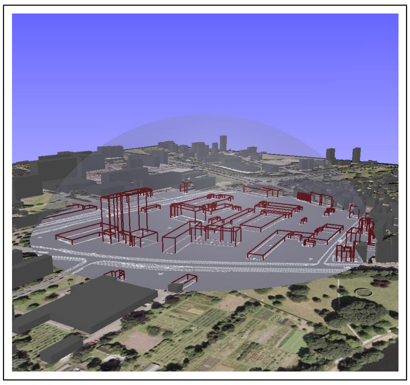
4.2 Urban Planning

Computer based planning of urban environments has attracted rising attention in the last decade. While in the past most city models have been built out of paper or are constructed using similar techniques, nowadays the computer is more often used to create and manipulate virtual city models. Most large cities in Germany do already have a virtual 3D city model which allows interactive exploration of the urban environment. One technique adequate for generating virtual 3D city models is a reconstruction based on cadastral data. By extruding buildings based on the information provided a construction of a virtual 3D city model is possible. In this subsection a short description is given how to apply 3D magic lenses to virtual city model reconstructed from cadastral datasets.

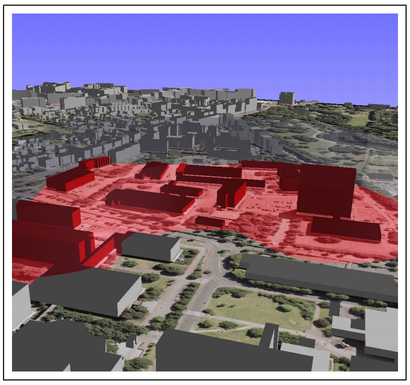
Usually cadastral datasets are organized in several layers representing different information, e.g., streets, recreation areas and



(a) building emphasizer



(b) wireframe buildings and detail reduction



(c) coverage of a wireless network

Figure 3: Application of magic lenses to a 3D city model.

buildings. By using 3D magic lenses, these layers can be visualized selectively. In Figure 3 the application of different 3D magic lenses within an interactive city planning environment is shown. The simple virtual city model is reconstructed from cadastral data of the city of Münster. In Figure 3(a) the same city model is shown with a *building emphasizer* 3D magic lens applied, highlighting all buildings intersecting the lens volume. The effect is achieved by associating a shape appearance with the lens. This shape appearance highlights all buildings, identifiable through an object name, by applying the color red during rendering. The same lens with a different visual appearance is shown in Figure 3(b). In this example the visual appearance assigned to the buildings has been extended by also applying wireframe mode in addition to highlighting. In this case the geometry of all roofs needs to be identifiable to include them within the exclude list associated with the lens. This is necessary, because otherwise the geometry of the roofs, which usually consists of several triangles, would have been rendered using wireframe mode as well, leading to undesired additional information. Besides the visual appearances assigned to the buildings and the roofs a visual appearance has been assigned to the ground intersecting the lens volume. Instead of applying the aerial photograph as ground texture, a visual representation of the cadastral data serves as the new texture for the ground plane. This leads to a different level of abstraction within the lens volume allowing to identify streets as well as building footprints on the ground. Figure 3(c) shows the application of a non-convex 3D magic lens to visualize the coverage of a wireless network.

Furthermore the application of 3D magic lenses can be used to visualize virtual city models with different levels-of-detail as they are described in Schilcher et al. (1999). Thus it would be possible to display buildings influenced by the lens using a detailed visualization, while buildings not within the scope of the lens can be rendered using a coarser level of detail.

4.3 Subsurface Visualization

Another application area for the 3D magic lens metaphor is the visualization of volumetric seismic datasets. One problem occurring in the visualization of information associated with volumetric data is that usually no insight view can be provided. Especially when navigating through a dense volume dataset the view of the camera will always be occluded. To avoid this problem we propose a camera lens which renders those parts of the volume dataset transparently that occlude the region of interest (see Figure 4). Within the lens volume a different threshold and a different color mapping has been applied for visualization. To allow better insights when rendering the parts in front of the lens volume we set the degree of transparency proportional to the amount of data occluding the region of interest. Thus we can get an occlusion free view to the region of interest by still maintaining contextual information displayed in front of the lens. To improve visualization of volumetric seismic datasets, 3D magic lenses can also be used as clipping volumes to reveal information hidden inside the dataset.

5 CONCLUSION

In this paper we have presented several application areas in geovisualization which potentially benefit from the use of the 3D magic lens metaphor. The described case studies have been realized based on the VRS graphics system which we have extended to support rendering of arbitrarily shaped 3D magic lenses at interactive frame rates. After giving a brief overview of related work we have classified the lenses used within this paper into two main types based on their usage and described different types of visual appearances which can be associated with a 3D magic lens. Based on some of these lenses we were able to propose a solution for focus and context visualization by emphasizing the region of interest within the lens volume by still maintaining contextual information outside the lens. In particular we have demonstrated the described focus and context visualization in three different application areas: virtual landscaping, urban planning as well as seismic visualization.

In the future we are going to extend the presented classification of different lens type to a complete taxonomy to better serve experts from different domains demanding focus and context visualization. Furthermore, we are going to provide a more flexible OpenGL based 3D magic lens rendering library, such that application developers can easily integrate the 3D magic lens metaphor in existing visualization applications not based on VRS.

6 ACKNOWLEDGMENTS

We would like to thank the reviewers of this paper for the fruitful comments.

This work was partly supported by grants from the Deutsche Forschungsgemeinschaft (DFG), Sonderforschungsbereich 656 MoBil Mnster, Germany (project Z1).

REFERENCES

Bier, E. A., Stone, M., Fishkin, K., Buxton, W., and Baudel, T., 1995. *A Taxonomy of See-Through Tools*, pp. 517–523. Morgan Kaufmann Publishers.

Bier, E. A., Stone, M. C., and Pier, K., 1997. Enhanced Illustration using Magic Lens Filters. Technical report, Xerox Palo Alto Research Center.

Bier, E. A., Stone, M. C., Pier, K., Buxton, W., and DeRose, T. D., 1993. Toolglass and Magic Lenses: The See-Through Interface. *Computer Graphics*, 27(Annual Conference Series):pp. 73–80.

Carpendale, M. S. T., Cowperthwaite, D. J., and Fraccia, F. D., 1997. Extending Distortion Viewing from 2D to 3D. *Computer Graphics and Applications*, 17:pp. 42–51.

Cignoni, P., Montani, C., and Scopigno, R., 1994. MagicSphere: an Insight Tool for 3D Data Visualization. *Computer Graphics Forum*, 13(3):pp. 317–328.

Curtis, M. P., Gerhardstein, A. C., and Howard, R., 1986. Interpretation of large 3-D Data Volumes. In *Expanded Abstracts of the Society of Exploration Geophysicists*, pp. 497–499.

Döllner, J., Baumann, K. A., and Hinrichs, K., 2000. Texturing Techniques for Terrain Visualization. In *Proceedings of the Conference on Visualization (VIS00)*, pp. 227–234. IEEE Press.

Fox, D., 1998. Composing Magic Lenses. In *Proceedings of the Conference on Human-Computer Interaction (CHI98)*, pp. 519–525. ACM Press.

Fröhlich, B., Barrass, S., Zehner, B., Plate, J., and Göbel, M., 1999. Exploring Geo-Scientific Data in Virtual Environments. In *Proceedings of the Conference on Visualization (VIS99)*, pp. 169–174. IEEE Press.

Fuhrmann, A. L. and Gröller, E., 1998. Real-Time Techniques for 3D Flow Visualization. In D. Ebert, H. Hagen, and H. Rushmeier, eds., *Proceedings of the Conference on Visualization (VIS98)*, pp. 305–312. IEEE Press.

IDELIX, 2002. Pliable Display Technology 3D. Technical report, IDELIX Software Incorporation.

North, C. and Schneiderman, B., 1997. A Taxonomy of Multiple Window Coordinations. Technical Report CS-TR-3854, Human-Computer Interaction Laboratory, University of Maryland.

Ropinski, T. and Hinrichs, K., 2004. Real-Time Rendering of 3D Magic Lenses having arbitrary convex Shapes. In *Journal of the International Winter School of Computer Graphics (WSCG04)*, pp. 379–386. UNION Agency - Science Press.

Schilcher, M., Guo, Z., Klaus, M., and Roschlaub, R., 1999. Aufbau von 3D-Stadtmodellen auf der Basis von 2D-GIS. *DGPF Heft*, 3:pp. 157–170.

Stoev, S. L., Schmalstieg, D., and Straßer, W., 2002. The Through-The-Lens Metaphor: Taxonomy and Application. In *Proceedings of the Conference on Virtual Reality (VR02)*, pp. 285–286. IEEE Press.

Stone, M., Fishkin, K., and Bier, E. A., 1994. The Movable Filter as a User Interface Tool. In *Proceedings of the Conference on Human-Computer Interaction (CHI94)*, pp. 306–312. ACM Press.

Viega, J., Conway, M. J., Williams, G., and Pausch, R., 1996. 3D Magic Lenses. In *Proceedings of the Symposium on User Interface Software and Technology (UIST96)*, pp. 51–58. ACM Press.

Wang Baldonado, M. Q., Woodruff, A., and Kuchinsky, A., 2000. Guidelines for using Multiple Views in Information Visualization. In *Proceedings of the Conference on Advanced Visual Interfaces (AVI00)*, pp. 110–119. ACM Press.

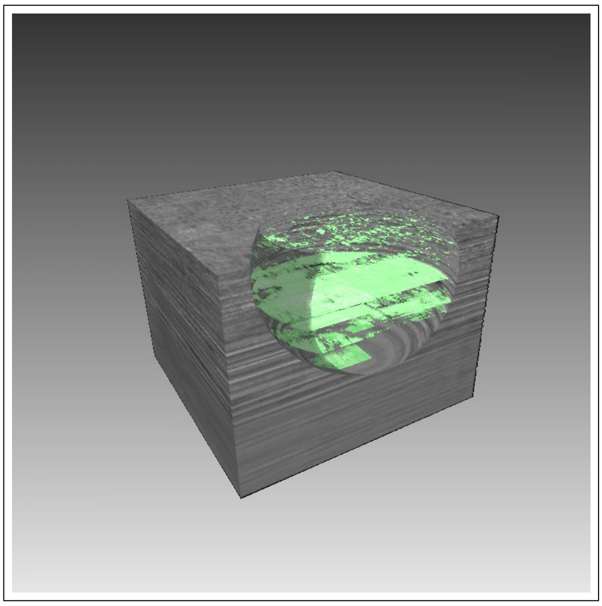


Figure 4: A 3D magic lens applied to a volumetric seismic dataset.