Through the Looking Glass:
The use of Lenses as an interface tool for Augmented Reality interfaces.

Julian Looser[1]
HIT Lab NZ
University of Canterbury, NZ

Mark Billinghurst[2]
HIT Lab NZ
University of Canterbury, NZ

Andy Cockburn[3]
Computer Science Department
University of Canterbury, NZ

Abstract

In this paper we present new interaction techniques for virtual environments. Based on an extension of 2D MagicLenses, we have developed techniques involving 3D lenses, information filtering and semantic zooming. These techniques provide users with a natural, tangible interface for selectively zooming in and out of specific areas of interest in an Augmented Reality scene. They use rapid and fluid animation to help users assimilate the relationship between views of detailed focus and global context. As well as supporting zooming, the technique is readily applied to semantic information filtering, in which only the pertinent information subtypes within a filtered region are shown. We describe our implementations, preliminary user feedback and future directions for this research.

Keywords

MagicLenses, Augmented Reality, interaction, transitional interfaces, semantic zooming.

1. Introduction

We have created a compelling implementation of 3D MagicLenses in an Augmented Reality (AR) setting. MagicLenses are semi-transparent user interface elements that apply transformations to whatever content lies beneath them [1]. We have developed novel techniques that employ these lenses to help users navigate, select objects and filter information in virtual environments. Our techniques are based around one universal tool: a hand-held magnifying glass.

AR interfaces fuse the real and virtual worlds together by accurately overlaying virtual content on a view of the real world. We have chosen this setting to implement our lenses for several reasons. Firstly, we have significant experience in this area. Secondly, we plan to extend our work to include collaboration, for which AR is a promising platform, as shown in [2], [3], [4] and others. Thirdly, AR interfaces promote the use of tangible props for interaction. Our lens tool is designed to mimic the feel of a real magnifying glass and is controlled via a tracked paddle.

At this stage we present our lens tools within a single-user environment, but discuss their exciting potential within collaborative and transitional virtual environments, spanning the continuum from reality to virtuality, such as the MagicBook.

The MagicBook is an example of one of our own collaborative, transitional interfaces [5]. It is a real book that allows its readers to smoothly transition between reality, augmented reality and virtual reality. The book can be read and enjoyed on its own, but

[1] email: julian.looser@hitlabnz.org
[2] email: mark.billinghurst@hitlabnz.org
[3] email: andy@cosc.canterbury.ac.nz

with the aid of a head-mounted display, 3D scenes pop out of the pages in an AR view (Figure 1a). At the press of a button, the reader can 'fly into' the scene and explore it from an immersive first-person perspective (Figure 1b).
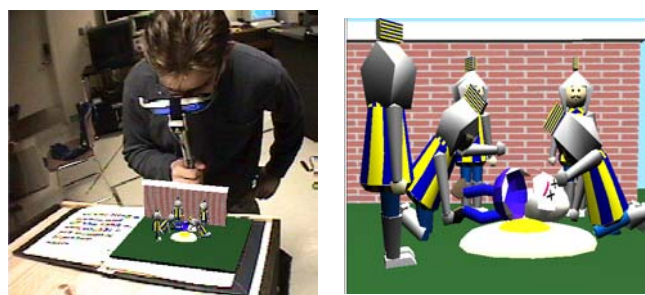
a) MagicBook AR.   b) Immersive VR View.
Figure 1: The MagicBook supports AR and VR.

Multiple users can participate simultaneously. To readers in AR, immersed users appear as small virtual characters within the scene. To each other, these users appear as life-sized characters in VR.

In this paper we describe our implementations of 3D MagicLenses and how they differ from, and extend, other work. We have created two applications to demonstrate the utility of our approach and report on the favourable feedback these interfaces have received. Furthermore, we discuss how our lens work can be exploited in the MagicBook interface to enhance its transitions between AR and VR.

2. Related Work

MagicLenses

MagicLenses were first introduced by Bier et al. [1] as a focus-and-context technique for traditional 2D interfaces. A MagicLens is a movable, semi-transparent user interface element that can change the representation of data shown beneath it. MagicLenses can be used for magnification as well as a wealth of other effects, such as previewing image effects (blur, for example) and level-of-detail (data through the lens is rendered at a higher resolution). Several lenses can be combined to produce composite effects where they intersect.

The MagicLens metaphor was extended to three dimensions by Viega et al.[6]. They implemented two types of 3D lens: a 'flat' lens that projected a volume of influence into the scene, and a volumetric lens that affected content falling within the space of a cube. Both approaches exploited hardware support for clipping planes which made it possible to divide the scene into lensed and un-lensed spaces in real-time.

1

Spatially Extended Anchor Mechanisms (SEAMs) are a navigation technique that provide portals between virtual environments [7]. A SEAM can be used to connect remote, virtual locations in such a way that the user can both look into the destination environment, and also venture there by moving through the SEAM. The ability to see into a different environment made it possible to implement 3D MagicLenses using the SEAMs framework. This was the approach taken by Fuhrmann and Gröller, who used both flat and volumetric lenses in their work on 3D flow visualisation techniques [8]. Flow data within the lens region was rendered in greater detail than the surrounding data, which could optionally be hidden completely. They claimed the lenses were useful for their visualisation purposes, but were difficult to control using a traditional mouse.

Stoev et al. [9] used MagicLenses in a virtual environment in which the view from a virtual camera was rendered onto a handheld pad. The virtual camera could be positioned at will within the scene, and various tools operated 'through-the-lens'; applying their effects onto the remote object whose image was selected on the pad. Objects in the lens view could be hidden to make these manipulations easier.

This prior research illustrates how MagicLenses have been used to provide an area of focus in a user interface while maintaining context. There are numerous other methods to this end, including distorted views, speed-dependent automatic zooming [10] and providing global views such as thumbnails and mini-maps.

Augmented Reality

As Milgram points out [11], interfaces can be classified according to the proportion of their content that is real versus how much is computer-generated, with Reality and Virtual Reality (VR) being the extreme cases (see Figure 2). Between these poles lie Mixed Reality (MR) interfaces, further classified as Augmented Reality (AR) and Augmented Virtuality (AV).
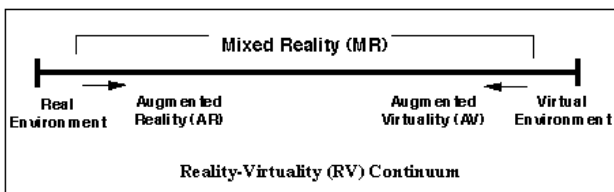

Figure 2: Milgram's Reality-Virtuality Continuum.

Augmented Reality interfaces are notable in that they involve the overlay of virtual imagery on the real world. AR has found use in a wide-range of applications, including manufacturing, medicine and entertainment.

Transitional Interfaces

Although there are many examples of interfaces that lie on the Reality-Virtuality continuum, few of these support transitions between reality, virtuality and points in-between.

One of the first interfaces to explore transitions in a fully immersive virtual environment was Worlds In Miniature (WIM) [12]. The user in a VR environment holds a small virtual version of the environment in which they are immersed. This provides the user with an exocentric view of their surroundings that can be used as a proxy for object selection and manipulation, and as an aid for navigation. This interface showed the value of transitions as manipulation and navigation tools, although in this case entirely in an immersive VR setting.

Koleva et al. investigated transitions between reality and virtual reality by creating real and virtual worlds connected by mixed-reality borders [13]. Their work focused on live performances in which the audience witnessed the illusion of seamless transitions which were facilitated by hidden ante-chambers and portals such as rain-curtains.

Kiyokawa's work on seamless viewmode switching is the most relevant to our own research. The interface allowed two users to collaborate at different scales around a virtual scene [14]. When both users shared a common life-sized body scale, the virtual scene was shown in an augmented reality view so that each user could see the world around them as well as the virtual imagery. When a user scaled themselves independently, the interface reverted to virtual reality in which each user saw the other as a correctly scaled avatar. Either user could initiate a transition that would smoothly adjust their body scale, and therefore transition between AR and VR. In their work, handheld magnetic trackers were used to provide gesture input and support the scaling between AR and VR modes.

3. Our 3D Lens Implementation

In this section we describe our implementation of 3D MagicLenses. We have implemented both flat and volumetric lenses in C++ using OpenGL. All our applications run in real-time on what we consider consumer-level hardware. An NVIDIA GeForce4 Ti-4800 SE graphics card was used during development but the code is not card-specific.

Rendering the Lenses

Volumetric Lenses

We render volumetric lenses by means of clipping planes using the method described by Viega et al. [6]. A clipping plane divides the scene into two half-spaces, one which is kept and one which is discarded. Modern graphics cards support clipping planes in hardware. There are six clipping planes that define the OpenGL view frustum as well as at least six additional planes that are available for general use by the programmer. Using six of these planes it is possible to construct a cube whose volume can be rendered differently to the rest of the scene (see Figure 3).
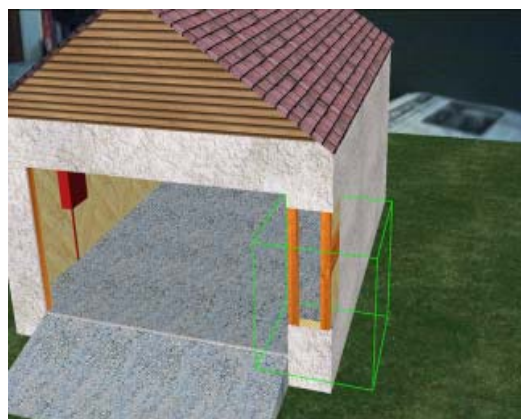

Figure 3: A volumetric lens configured to render only the internal framing of the building.

a) Stencil buffer contents. White indicates the area of the lens.

b) The area outside the lens is rendered.

c) The area within the lens is rendered with some effect. In this case, the shell of the building is removed, exposing the framing inside.

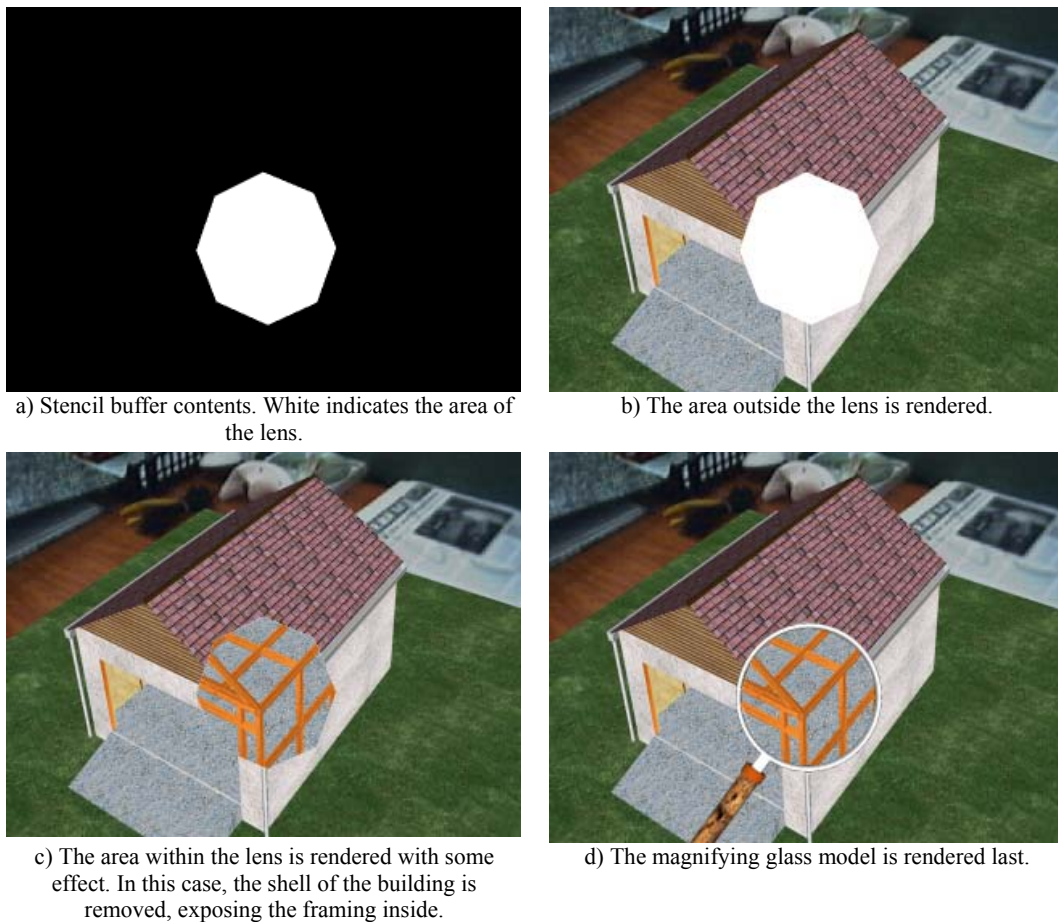d) The magnifying glass model is rendered last.

Figure 4: The process of rendering a flat lens.

Rendering the content inside the cube is simple. All planes are enabled such that they discard all regions outside the cube. The scene is then rendered with the desired effect applied. This may involve hiding certain objects, or using a particular rendering style such as wireframe. Rendering the content outside the cube is somewhat more complicated. Simply reversing the direction of the clipping planes will not invert the rendered areas. Clipping planes in OpenGL extend to infinity so that two parallel, outward facing clipping planes will clip the entire scene (see Figure 5). To overcome this problem, the scene must be rendered six times, once as each individual clipping plane is active on its own.

a) Inward facing planes. Object is clipped.

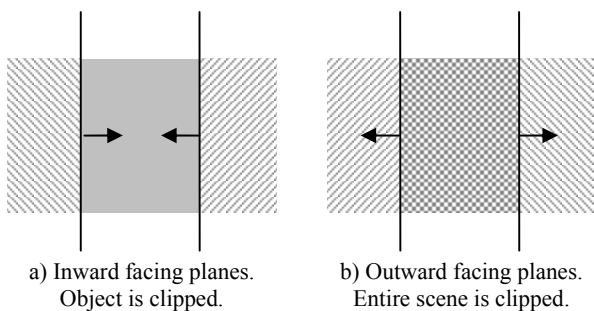b) Outward facing planes. Entire scene is clipped.

Figure 5: Rendering using clipping planes. The arrows indicate the side of the plane that is kept. Diagonally shaded areas are clipped while solid areas remain. (Figure adapted from [6]).

Fuhrmann and Gröller describe a technique without this inefficiency [8], but it results in geometry that should be visible behind the lens not being rendered. In our applications the inefficiency has no noticeable effect on performance. However, we are currently using models with low polygon counts and as scene complexity increases performance will degrade exponentially.

Flat Lenses

As mentioned, Viega et al. [6] showed how to implement both volumetric and flat 3D MagicLenses. Our method for creating flat lenses differs substantially from that of [6], and is more closely related to that of [7]. We created flat lenses by using the OpenGL stencil buffer to mask out lensed and un-lensed areas of the screen. The mask is created by rendering the lens object itself into the stencil buffer resulting in a value of 1 where the lens exists and a value of 0 elsewhere (Figure 4a). The scene is then rendered normally in areas equal to 0 (Figure 4b) and with some effect applied in areas equal to 1 (Figure 4c). Finally the lens itself and its accompanying handle are drawn on top (Figure 4d).

This technique made the lens more flexible than using clipping planes, where the number of available planes limits the shape of the lens, typically to a quadrangle. Our method supports lenses of any shape and initially we have used a circular lens mounted inside a magnifying glass model.

We feel that the magnifying glass is a fitting tool in our research as it is universally recognised as a tool for investigation; users understand that they should peer through the lens to examine things that cannot be seen with the naked eye. With a virtual magnifying glass we can extend this notion to allow the user to see through objects and to see the objects represented differently through the lens.

4. Augmented Reality Interaction Techniques Afforded by Lenses

Focusing for now on our flat lens implementation, we have developed ways in which the lens can be used to accomplish a variety of fundamental interaction tasks.

Magnification: As a tool for examining distant objects up close, or close objects in greater detail.

Object Selection and Manipulation: As a tool for selecting and manipulating virtual objects in view.

Information Filtering: As a tool for filtering the information shown in the AR and VR views, either by selectively hiding content, or adjusting its representation.

Here we describe these techniques and possible extensions to them.

Using the Lens for Magnification

The virtual lens can be used in the same way as one would expect to use a real lens: for magnification. However, in the real-world, when we use a magnifying glass we can only control the scale of what we see through the lens. In a virtual environment, we have the ability to scale the surrounding environment as well. At the press of a button, the user can initiate a smooth zoom of the surrounding scene to match the magnification they have selected through the lens. This technique is similar to Kiyokawa's seamless viewmode switching [14], but rather than having two users who can scale themselves independently around a 3D scene and also transition to their partners scale, there is a single user in control of both scale settings.

This mode of interaction would be useful when examining a model, such as a virtual historical artifact. When a particular point of interest was discovered, the researcher could use their magnifier to zoom and study that point. If the surrounding area also appeared to be interesting, then the researcher could effortlessly scale the entire scene to the selected zoom level.

Using the Lens for Object Selection and Manipulation

The lens defines an area of focus within the scene. We can base object selection on whether an object lies partly or completely within this area. This is essentially ray-casting if we select the objects targeted by the center of the lens, or cone-casting [15] if we select all objects within the lens space. However, because the user peers through the lens to make the selection, we predict that selection will be easier than with conventional implementations of either of these techniques.

Once an object is selected, we can use the lens to perform a variety of operations on the object. For example, we could bind the object's scale to the magnification of the lens, such that as

the user magnifies, it is now only the selected object that changes size. Similarly, we could bind the object's position to the lens so that the user could move the object to a new location simply by looking at that location through the lens. This technique could be coupled with a cloning operation so that multiple instances of the object could be 'stamped' throughout the scene.

Using the Lens for Information Filtering

One of the fundamental characteristics of a MagicLens is the ability to present a different representation of the underlying data. Our lenses can reduce the complexity of a user's view by removing data that is irrelevant to them during their current task. For example, a complete model of a building might contain 3D data for dozens of different systems, such as electrical wiring, water supply and fire-escapes. It is unlikely that a single user will require, or be able to comprehend, all datasets at once, so some form of filtering is required. Using the lens, the user can select which datasets are shown both inside and outside the lens area. The filtering criteria can be changed in real-time so that different aspects of the data can be explored.

An obvious use of this ability is to cut away the surface of an object to expose its inner workings. This method of viewing is the foundation of the immensely popular *Incredible Cross-Sections* series of books illustrated by Stephen Biesty [16]. These books contain cutaway drawings of historical buildings, advanced machines and many other interesting items. We believe that our augmented reality lenses are the ideal platform for advancing this popular concept into an interactive, three-dimensional setting.

Julier et al. tackled the problem of clutter in augmented reality interfaces and developed an algorithm for automatically filtering information [17]. Another approach is to dynamically alter the view based on the current magnification. This technique is known as *semantic zooming* [18]. As the user magnifies a particular area, additional information specific to that area can be incorporated into the view. Showing this data all the time would clutter the interface so it is only added in as it becomes relevant.

Using the Lens Combinations

Our lens operations can be chained together in interesting ways to accomplish complex tasks. For example, a lens could be used to filter a dataset to show only the objects of interest and then we could change to a selection mode and use the same lens to select one of the filtered objects. Similarly, once we have selected an object, the lens magnification tool could be used to zoom the view so that the object is at the desired scale.

Sample Applications

In order to explore how lens techniques could be used in an AR interface we created two sample applications: a globe visualisation and a virtual house demonstration.

In both of these demonstrations the user held a virtual lens over an AR view of a virtual model. The AR tracking was provided by the ARToolKit library [19], computer vision software which can calculate a real camera position from a set of one or more fiducial markers.

a) Base tracking grid or markers.

b) Handheld trackball with attached marker.
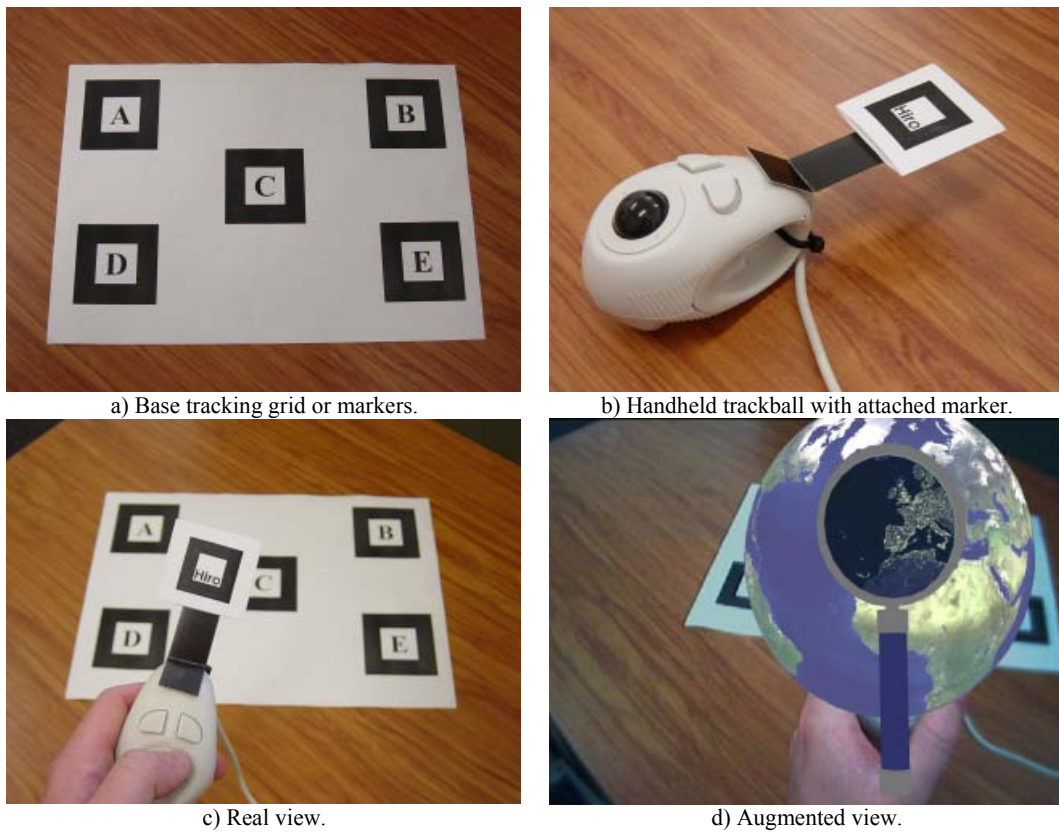

c) Real view.

d) Augmented view.

Figure 6: Tracking arrangement.

Using ARToolKit, the 3D scene is rendered on top of a large grid of markers. The lens is bound to a smaller marker attached to a handheld trackball. This technique is known as *paddle interaction* and is a common approach in AR interfaces, [20] for example. The user can configure the effect they see applied through the lens using the trackball's controls. This tracking and input arrangement is shown in Figure 6.

We use a video see-through AR technique which means that the user wears a virtual reality headset with a small video camera attached at the approximate position of their eyes. Each frame from the camera is processed by a computer which overlays the 3D graphics on the image. The image is then displayed on the user's headset. The headset used in our demonstrations was a Cy-Visor DH-4400VP and the camera used was a Creative Webcam 5 USB.

Globe Demonstration

In the globe demonstration, users could cycle the lens through a variety of worldwide datasets while maintaining a default view outside the lens. This application presents a novel way to visualise the wealth of global information available. For example, Figure 7a shows chlorophyll data [21], Figure 7b shows city light data [22] and Figure 7c shows NASA's Blue

a) Chlorophyll data.
Credit: Provided by the SeaWiFS Project, NASA/Goddard Space Flight Center and ORBIMAGE.

b) The Earth at night.
Credit: C. Mayhew and R. Simmon (NASA/GSFC). NOAA/NGDC, DMSP Digital Archive.

c) NASA Blue Marble imagery.
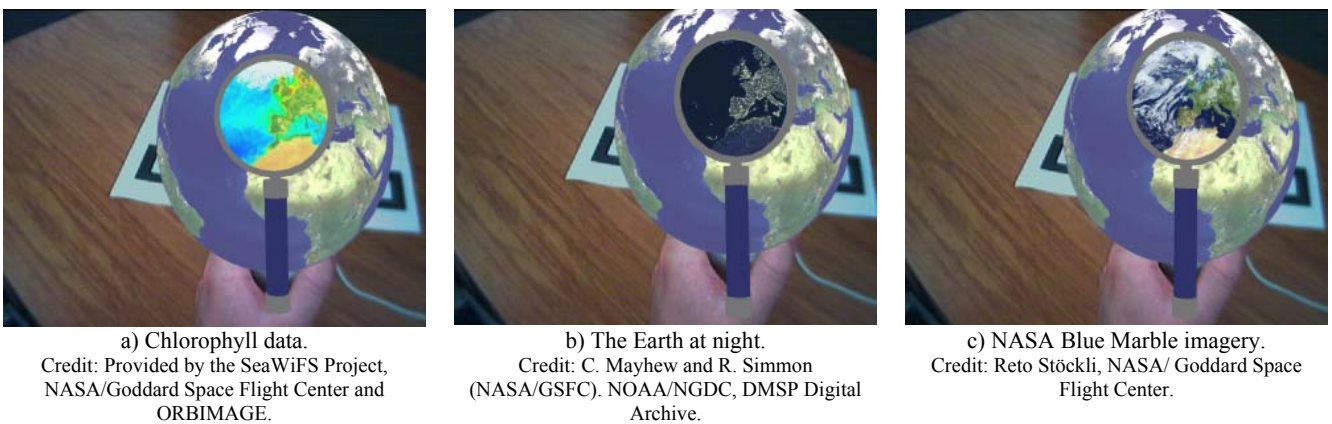Credit: Reto Stöckli, NASA/ Goddard Space Flight Center.

Figure 7: Examining various datasets on the globe. Each picture above illustrates a different dataset but the same geographical location. (Images may be difficult to discern without colour.)

Marble image: "the most detailed true-color image of the entire Earth to date" [23]. There are literally dozens of additional data sets that can be viewed in this way. Because standard maps are centered on the prime meridian (the north-south line through Greenwich, 0º longitude), it is a simple task to import new data into the globe application. When the user has found a particularly interesting dataset, they can apply it globally so that it becomes the context rather than the focus.

House Demonstration

In the house demonstration, various components of a virtual house model can be enabled or disabled through the lens. For example, all parts of the house other than the internal wooden framing can be turned off so that through the lens the user sees the frame while outside the lens the complete house remains (see Figure 8).

In practice, such a technique could allow people with diverse skills and interests to efficiently collaborate around a design project, such as a house or piece of hardware. Typically, a builder would be interested in the structural details such as framing and materials, as well as information relating to the components and the order of construction. On the other hand, a decorator may wish to be able to peer into the building and see an entirely different view; one where furniture is displayed and realistic lighting is rendered. Such a view would allow them to make sensible choices as to how to decorate the building. Many other views are possible for architects, real-estate agents, electricians and so forth. Each view benefits from the focus and context nature of the lens and illustrates the additional advantage of information filtering.

The ability to transition into a VR view allows users to explore the environment from a first-person perspective, while still in possession of their lens tool. Continuing the building scenario from above, each user could navigate around the building while still in possession of their lens tool. From this perspective they could examine the interior of the building and still benefit from the information filtering abilities of the lens.

User Feedback

Several people have used the applications and initial user feedback has been very encouraging. Users from a variety of backgrounds have described the systems as feeling natural, both in terms of using the tangible prop as a magnifying glass and the virtual content filtering. Several users have commented on how applications like the globe demonstration would be perfect educational tools, a sentiment we wholeheartedly agree with.

5. Discussion: Using Lenses in Transitional AR Interfaces

Transitional interfaces allow users to move between points on the Reality-Virtuality continuum (see Figure 2). The MagicBook interface currently supports a smooth, but uncontrollable journey from AR to VR. We believe that our work with 3D lenses can be used to enhance this transition into a more powerful tool. Ideally, the user will be able to select an arbitrary scale with which to view the scene before them.

The user first focuses the lens on the item of interest and then selects their preferred scale using the magnifier. When the image in the lens matches their intended scale, the user presses a button, at which stage the entire scene seamlessly animates, either by growing or shrinking, to match that scale. If the user has selected a scale other than 1:1, then the interface ceases to operate in augmented reality and instead presents an entirely virtual representation of the scene. In this virtual reality, the scene is no longer treated as an object to be examined, but rather an environment to be explored. The user can freely fly around the virtual world or walk around it, depending on their currently selected scale.

6. Conclusions and Future Work

We have implemented flat and volumetric 3D MagicLenses within an augmented reality setting. The lenses allow users to magnify content, select and manipulate objects, and customise their view in a variety of useful ways. Although we plan to implement more techniques based around the lens, our current techniques form a useful set of tools. We have demonstrated two compelling applications of this technology: a globe for visualising and comparing global datasets, and a house model that shows how the lens can reduce the complexity of a scene and can be used to highlight particular features.
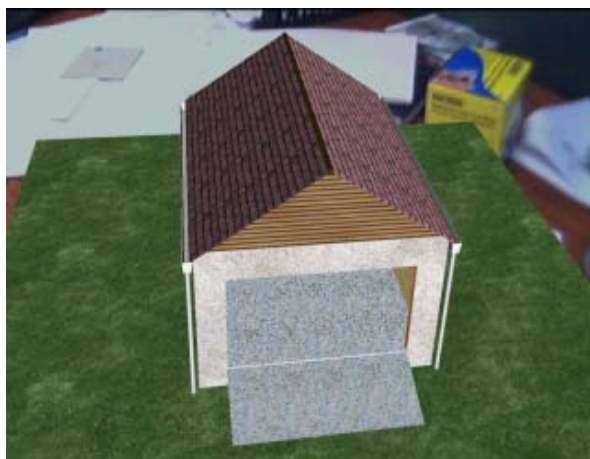
a) AR Scene   b) Filtered View

Figure 8: The house demonstration with and without the lens visible.

Informal feedback has told us that users find our interfaces fascinating. We suggest that there is a significant opportunity to exploit this technology in education and entertainment.

We plan a substantial amount of further work in this area.

- We intend to integrate our new lens techniques with the existing MagicBook interface, and to explore how we can make transitions between AR and VR more configurable.

- We plan to utilise the lens techniques described in this paper in the visualisation of more practical data such as real geographical datasets. Using these new applications we will run more rigorous user studies and implement further interaction techniques based on the lenses.

- We plan to progressively incorporate more of the original MagicLens concepts into our implementation. For example, we wish to be able to combine multiple lenses in augmented reality.

We believe MagicLenses have a lot to offer within virtual environments, particular in augmented reality, where the use of a tangible magnifying tool makes the MagicLens metaphor all the more powerful.

Acknowledgements

References

[1]   E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose, "Toolglass and MagicLenses: The See Through Interface," Proceedings of Siggraph 93, pp. 73-80, 1993.

[2]   G. Reitmayr and D. Schmalstieg, "Mobile collaborative augmented reality," ISAR.

[3]   M. Billinghurst and H. Kato, "Collaborative Mixed Reality," In Proceedings of International Symposium on Mixed Reality, pp. 261-284, 1999.

[4]   D. Schmalstieg, A. Fuhrmann, G. Hesina, Z. Szalavári, L. M. Encarnação, M. Gervautz, and W. Purgathofer, "The Studierstube Augmented Reality Project," Presence, vol. 11, pp. 33-54, 2002.

[5]   M. Billinghurst, H. Kato, and I. Poupyrev, "The MagicBook - Moving Seamlessly between Reality and Virtuality," IEEE Computer Graphics and Applications, vol. 21, pp. 6-8, 2001.

[6]   J. Viega, M. J. Conway, G. Williams, and R. Paush, "3D Magic Lenses," Proceedings of the 9th annual ACM symposium on User interface software and technology, pp. 51-58, 1996.

[7]   D. Schmalstieg and G. Schaufler, "Sewing worlds together with SEAMS: A mechanism to construct complex virtual environments," Presence - Teleoperators and Virtual Environments, vol. 8, pp. 449-461, 1999.

[8]   A. L. Fuhrmann and E. Gröller, "Real-Time Techniques for 3D Flow Visualization," IEEE Visualization '98, pp. 305-312, 1998.

[9]   S. L. Stoev, D. Schmalstieg, and W. Straßer, "The Through-the-Lens Metaphor: Taxonomy and Application," IEEE Virtual Reality Conference 2002, pp. 285-286, 2002.

[10]   T. Igarashi and K. Hinckley, "Speed-Dependent Automatic Zooming for browsing large documents," UIST, pp. 139-148, 2000.

[11]   P. Milgram and F. Kishino, "Augmented Reality: A Class of Displays on the Reality-virtuality Continuum," SPIE, Telemanipulator and Telepresence Technologies, vol. 2351, pp. 42-48, 1994.

[12]   R. Stoakley, M. J. Conway, and R. Pausch, "Virtual Reality on a WIM: Interactive Worlds in Miniature," Proceedings of CHI'95 Conference on Human Factors in Computing Systems, 1995.

[13]   B. Koleva, H. Schnädelbach, S. Benford, and C. Greenhalgh, "Traversable interfaces between real and virtual worlds," Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 233-240, 2000.

[14]   K. Kiyokawa, H. Takemura, and N. Yokoya, "A Collaboration Support Technique by Integrating a Shared Virtual Reality and a Shared Augmented Reality," TVRSJ, 1999.

[15]   D. A. Bowman and L. F. Hodges, "User Interface Constraints for Immersive Virtual Environment Applications," Graphics, Visualization, and Usability Center 1995.

[16]   J. R. H. Platt and S. Biesty, Incredible Cross-Sections: Knopf Books for Young Readers, 1992.

[17]   S. Julier, M. Lanzagorta, Y. Baillot, L. Rosenblum, S. Feiner, and T. Höllerer, "Information Filtering for Mobile Augmented Reality," Internation Symposium on Augmented Reality 2000, pp. 3-11, 2000.

[18]   B. B. Bederson and J. D. Hollan, "Pad++: A Zoomable Graphical Interface System," Conference companion on Human factors in computing systems, pp. 23-24, 1995.

[19]   ARToolKit,   "ARToolKit,"   in http://www.hitl.washington.edu/artoolkit/.

[20]   N. Hedley, M. Billinghurst, L. Postner, R. May, and H. Kato, "Explorations in the use of Augmented Reality for Geographic Visualization," Presence, vol. 11, pp. 119-133, 2001.

[21]   SeaWiFS, "SeaWiFS Image Gallery," in http://seawifs.gsfc.nasa.gov/SEAWIFS/IMAGES/SEAWIFS_GALLERY.html.

[22]   NASA, "Astronomy Picture of the Day, August 10, 2002," in http://antwrp.gsfc.nasa.gov/apod/ap020810.html, 2002.

[23]   NASA, "The Blue Marble," in http://earthobservatory.nasa.gov/Newsroom/BlueMarble/.