

# Boom Chameleon: Simultaneous capture of 3D viewpoint, voice and gesture annotations on a spatially-aware display

Michael Tsang<sup>2,1</sup>, George W. Fitzmaurice<sup>1,2</sup>, Gordon Kurtenbach<sup>1,2</sup>, Azam Khan<sup>1,2</sup>, Bill Buxton<sup>1,2</sup>

<sup>1</sup>Alias|wavefront  
210 King Street East  
Toronto, Ontario  
Canada M5A 1J7

{gf | gordo | akhan | buxton}@aw.sgi.com

<sup>2</sup>Dept. of Computer Science  
University of Toronto  
Toronto, Ontario  
Canada M5S 3G4

{mtsang | gf | gordo | buxton}@dgp.toronto.edu

## ABSTRACT

We introduce the *Boom Chameleon*, a novel input/output device consisting primarily of a flat panel display mounted on a mechanical boom that is tracked in 3D space. The display acts as a physical window into 3D virtual environments in which a direct one-to-one spatial mapping between real and virtual space is preserved. The Boom Chameleon is augmented with a touch screen overlaid on the display and a microphone/speaker combination. We present a 3D-annotation application that exploits these unique characteristics in order to simultaneously capture viewpoint, voice and gesture information. We discuss design issues and present results of an informal user study on the device and annotation application. Our results show that our Boom Chameleon system has the potential to be an effective, easy to learn and operate 3D design review system.

**KEYWORDS:** 3D navigation, annotation, gesture, voice, spatially-aware display.

## INTRODUCTION

In the realm of digital media, three-dimensional data has come of age - The use of 3D models is prolific, ranging from industrial design to scientific visualization. Tools for the creation and modification of complex models have become incredibly sophisticated and powerful. Yet the technology for annotating these models has been left far behind, resulting in a gaping discontinuity in the high-technology workflow. Due to the widespread use of 3D data, an effective annotation system is likely to prove extremely useful in industry and offers exciting design opportunities for creating a compelling and interactive user experience.

From our work with automotive designers, they have stated that in current practice, reviews and critiques of

digital three-dimensional models can be awkward and unwieldy. Take for example a representative automotive design review where managers and designers are grouped together in front of a large display: The manager critiques the design from various viewpoints but must rely on a technician to operate the interface and move the 3D model to various viewpoints. Meanwhile, the designers are furiously scribbling the manager's comments into a notebook, perhaps marking-up images of the 3D model or making quick sketches to capture the manager's intentions.

Managers and designers have reported three major problems with this situation:

*Viewpoint control* – Managers and designers feel frustrated because they cannot directly control the 3D viewpoint. This contrasts with the evaluation of a physical 3D model where anyone is free to move around the model and direct the group's attention.

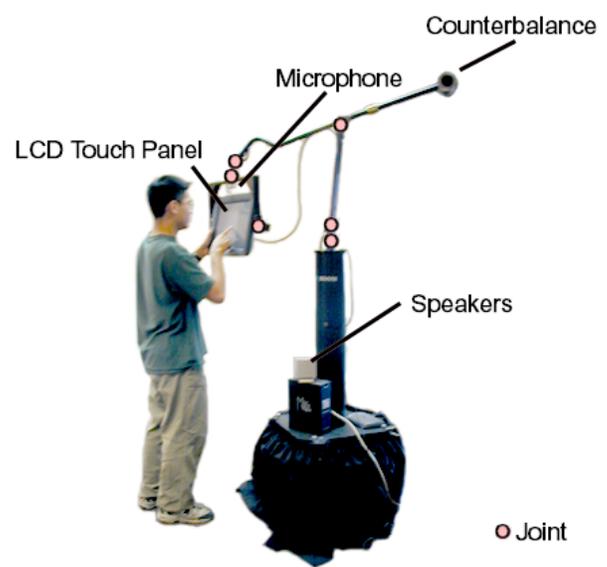


Figure 1. The Boom Chameleon.

LEAVE BLANK THE LAST 2.5cm  
OF THE LEFT COLUMN  
ON THE FIRST PAGE  
FOR US TO PUT IN  
THE COPYRIGHT NOTICE!

*Information capture* - Designers feel that critique information is lost because attention is divided between recording the annotation and participating in the discussion.

*Shared view* - Both managers and designers feel misunderstandings could be reduced if notes and annotations could be created and viewed by all parties simultaneously.

Outside the realm of 3D visualization, there is a substantial body of literature on supporting reviews and critiques through digital annotations on 2D assets such as documents and images. An emerging approach is to take as input existing workpractice annotation styles such as marking-up documents or simply speaking and captures them. For example, the Wang Freestyle system [17] is an early example of such an annotation system. A key to the success of these systems is in making the capture of the annotations minimally intrusive. One approach to minimizing intrusion is to produce a system that allows for the simultaneous capture of different types of annotation information. In effect, these systems constantly record all data streams removing the burden of the user having to explicitly engage different annotation tools.

In terms of 3D visualization, over the past few years we have been experimenting with a specialized hardware device known as the Boom Chameleon (see Figure 1). The Boom Chameleon is a “spatially-aware” device [9, 10] where the display acts as a window onto a virtual world, where moving the display in space changes a viewpoint into the virtual world. This concept can also be thought of as a “video camera” metaphor.



Figure 2. Navigating with the Boom Chameleon.

We have found that the Boom Chameleon makes some forms of 3D navigation easy. Viewpoint controls typically found in commercial computer graphics applications have discrete functions for zooming, panning, tumbling, etc. In contrast, the Boom Chameleon replaces all of these

operations with simple physical manipulation of the display. In essence, the virtual model remains fixed in physical space while the user moves the display around in that physical space. This allows an untrained user to move around a 3D model and inspect it from different viewpoints. As Figure 2 shows, when a user moves to the front of the space, the front of the virtual car is viewable, moving to the side shows the corresponding side of the car, and moving closer to the center zooms in to a close-up of the car. To return to a particular view, the user simply returns to the same physical point in space. Because of these characteristics, the Boom Chameleon is very well suited for allowing non-technical users to control the viewpoint when performing reviews and critiques of 3D models.

Given these properties of the Boom Chameleon and the research on simultaneous capture of 2D annotations and speech, this paper explores combining these approaches to address the effective review and critiques of 3D digital models. The intention is that this approach may address the three major problems brought forward by the managers and designers. First, the ease of use and operation of the Boom Chameleon will address the viewpoint problem. Second, information capture techniques from 2D annotation schemes can be adapted to address the information capture problem. Lastly, the Chameleon display will address the shared view problem by serving as a common viewing and annotation space.

In this paper we first review the related research and then more formally present the design of our system and the implementation details. We next describe the results of several informal user studies. Lastly, we discuss lessons learned, design insights and the general feasibility of this approach.

## RELATED RESEARCH

### Chameleon Navigation

The Boom Chameleon system is based upon the “window in hand” metaphor introduced by Fitzmaurice [10, 9]. Perhaps the most closely related system is the room-scale Art+Com virtual car installation [1]. Others take advantage of a handheld spatially-aware display to navigate 2D virtual information such as newspapers [29]. The window in the hand metaphor has also been augmented with video cameras on the handheld device to view the real world with computer graphics superimposed on the display [27, 20].

The WindowsVR system [36] is a close cousin of the Boom Chameleon. WindowsVR consists of a touch sensitive LCD display which is hung from an overhead arm using a retractable cable. The display comes in two configurations: In the first configuration, only the

orientation of the display is sensed. For translation, the user operates two joysticks, mounted on each side of the LCD. In contrast to the Boom Chameleon, this breaks the 1-1 spatial correspondence between the physical and virtual spaces as the user must navigate using a “driving” or “flying” metaphor. In the second configuration, both the orientation and translation are sensed but because the display is suspended from a cable and hence only counterbalanced in one direction, it can be cumbersome to operate.

Other work has looked at understanding the relationship between eye, hand and display when designing navigation and interaction techniques. This work highlights Chameleon’s unique properties in relation to other VR approaches, such as head-mounted-displays, Caves, and the standard workstation configuration [5].

### **2D Annotation**

There is a wide range of research concerning annotation of 2D media such as images, documents, and video. The most relevant work from this area concerns “free-form” hand drawn mark-up and voice annotations in a modeless fashion. Although the earliest work that makes use of voice input and deictic gesturing is the “Put-That-There” system [3], the Wang FreeStyle [17] system is one of the earliest to use voice input and gesture in an annotation context, capturing voice and ink mark-up for documents. Other systems have followed similar approaches for meeting capture [22, 25], collaborative writing [12], video annotation [18, 32], note taking [34, 35], and voicemail [33, 11]. Studies of annotation in tradition media such as paper documents suggest that a free-form strategy is attractive since people show a great diversity in how they annotate [24, 21]. Furthermore, there has been evidence that spoken annotations can be richer than textual annotations [6].

### **3D Annotation**

The notion of annotating 3D scenes has been explored in many different forms. In terms of commercial systems, these are generally designed for traditional monitor, mouse and keyboard configurations and typically use a fairly structured and modal approach to the interface. For example, the SGI – IRIS Annotator [14] allows annotation of 3D models with several media types from fixed viewpoints, employing explicit modalities for input including keyboard text, audio and video. Other research systems have explored similar methods for embedding structured data into the 3D scene. Loughlin and Hughes [19] use virtual post-it text notes and some systems [16, 13] use structured graphics such as pointer arrows or icons to indicate annotations in the 3D scene. Researchers have observed that these structured inputs are not always sufficient [15] and the ability to free-form draw would be beneficial. Early explorations have been performed on the ability to free-form draw annotations directly on 3D

models [15, 7] and onto 2D drawing planes in a 3D scene [16, 4, 30] and show promise. Our work continues to develop along this research vein.

The Virtual Notepad [26] system uses a head-mounted display and a spatially tracked tablet to control the location of a virtual writing surface in an immersive world. Annotations may be placed in the virtual world by positioning the tablet and writing on its surface. The Boom Chameleon differs from this work in that our display is not head mounted and, consequently, our combination of display/touchscreen allows for a different set of user affordances.

The main contribution of this paper over prior work is combining a spatially-aware display with the ability to capture voice, gesture and viewpoint information in a continuous, free-form fashion.

### **DESIGN PHILOSOPHY**

Our design philosophy for the Boom Chameleon annotation system is to create a system in which annotations can be created with minimal overhead. We also wish to minimize the amount of planning ahead that a user must consider. To accomplish this, we capture different types of annotation data without requiring a user to make explicit actions for each data type. This is achieved by constantly recording all streams of annotation data. We believe that this approach will allow a richer, more useful set of captured annotations when compared with a more structured and modal approach to annotation entry.

### **BOOM CHAMELEON**

The Boom Chameleon is a specialized input/output device that is a modified version of the Fakespace BOOM3C. The BOOM3C consists of a pair of stereo goggles mounted on a position and orientation-sensing “boom armature” [8]. The goggles were replaced with a SGI 15” (16:9 aspect ratio) LCD display panel (see Figure 1). The boom was subsequently re-balanced in order to have the display float weightlessly in space. With this configuration, the display has a 10-foot spherical range of motion around the center of the boom.

Replacing the goggles with a touch screen dramatically affects the affordances of the device [5]. As a result, users can now look at virtual objects while carrying on a face-to-face conversation with colleagues and see each other’s hand gestures relative to the display.

To facilitate touch input, a customized Elo touch screen was overlaid onto the LCD display. This allows the hands to simultaneously operate two modes of input: one hand may hold and control the position of the display, while the other hand utilizes the touch screen.

To record voice, we attached a podium microphone to the top of the display frame. Speakers were employed for sound output.

### ANNOTATION APPLICATION

To facilitate design reviews and critiques, we have developed an annotation application to take advantage of the simple navigation metaphor of the Chameleon.

Our graphical interface was designed to allow for a wide range of possible grip and hand positions (see Figure 3). The wide button strips running along the left and right sides of the display are both simple and accessible using thumbs from a wide variety of grips. In addition, there are large push buttons along the top of the screen.



Figure 3. Hand grip and widget layout.

### Viewpoint and Voice Capture

From the user's perspective, the view reacts immediately in response to changes in the position and orientation of the display. The microphone placed above the display is well positioned to capture speech from the primary annotator and discussions with colleagues who are also viewing the display. When the Record button is active, both viewpoint and voice information are saved as separate data streams, ready for immediate playback.

Since some ambient sounds within the room are also picked-up, as with any audio recording situation, privacy issues need to be considered. While we indicate when a recording session is in progress through a large red button on the display, a secondary red light could be added to the top of the Boom Chameleon's central post to signal to nearby colleagues that a recording session is in progress.

### Gesture Capture

Gestures that are supported by our annotation system can be grouped into two categories: non-marking and marking. In the course of a normal conversation, a significant number of deictic gestures may be made. Deixis is a well-studied form of natural dialogue that allows the user to efficiently express their ideas by combining voice and gesture information in a single

message. This is particularly useful when dealing with spatial data as in our 3D object inspection.

*Flashlight.* In order to support non-permanent deictic gesture, we implemented a mode that simulates a flashlight on the tip of the finger which can be used to highlight locations on a 3D model (see Figure 4a). The 'virtual flashlight' is invisible, but the effect is that a spotlight appears on the model from the direction that the user is pointing. We found through preliminary tests that often the flashlight was not precise enough for detailed pointing tasks. Consequently, we indicate the precise location of interest by using a small ball that floats above the model. Since the ball is directly above the point of interest, a reviewer of the annotation can view the recording from different angles, and still have the gesture aligned appropriately on the model. It is also possible to infer the viewpoint of the annotator from the shape of the spotlight.

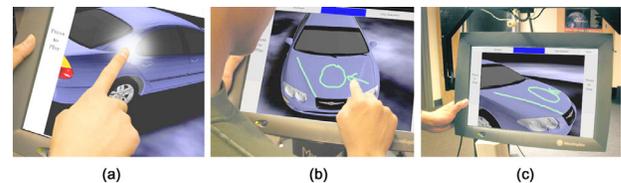


Figure 4. (a) Flashlight feature (b) drawing on object (c) inspecting drawing from new view.

To implement Flashlight pointing mode, we place the flashlight above the model one tenth of the distance between the location of the screen and the model. This combination allows the user to highlight a larger area of the model when farther away, smaller regions when closer, and the small sphere allows exact location of features when the user is very close.

*Pen - Drawing on the 3D object.* We also provide an ability to draw directly on 3D objects in the scene. This allows for annotations to be readily viewable from multiple viewpoints (see Figure 4b,c). Although Pen mode is the default, either Flashlight or Pen mode and can be rapidly toggled by selecting the appropriate button at the top of the screen (see Figure 3a,b).

*Snapshots.* Snapshots in our system are flat 2D images floating in the 3D scene (see Figure 5). This differs from the snapshot approach in [28] where snapshots are also floating in 3D space but are windows into 3D volumes. In our system, to create a snapshot the user clicks on the "Take snapshot" command (see Figure 3c) that causes an image of the current viewpoint to be captured and positioned in front of the camera. This also temporarily locks the viewpoint; enabling the LCD screen to be used as a writing pad. The pen can be used to write directly on the snapshot and the system continues to capture voice annotations. When finished, the user presses the "Done Snapshot" button, releasing the snapshot image.

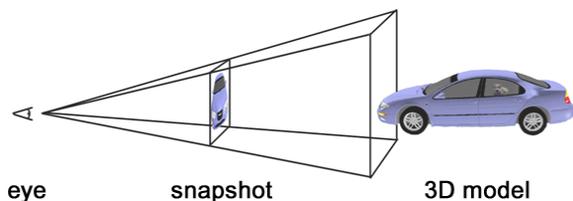


Figure 5. Snapshot of active view.

The snapshot capability is useful for a variety of situations. First, while the pen can be used to draw on 3D objects, there is no mechanism for “drawing where there are no objects”. This can be an especially acute problem in a sparse scene. For example, it is difficult to circle a group of atoms in a molecule (see Figure 6). The snapshot allows drawing “in space” since the markings appear on the surface of the snapshot. Second, because the viewpoint is movable while drawing, moving the display also moves the pen tip relative to the scene. We term this phenomenon “ink drift”. Normally, the viewpoint can be held steady enough to make rough marks, however, ink drift becomes a problem when exact precision is needed. The frozen viewpoint of the snapshot feature helps to alleviate this problem. Third, some viewpoints are hard and awkward to reach. Taking a snapshot allows users to grab a difficult viewpoint and reposition the display to a more comfortable viewing and writing posture. Finally, drawing on an object can be effective in many situations, however, users may mistakenly assume the marks are being drawn on a plane in screen-space rather than object space. This results in marks looking correct from the original drawing viewpoint but extremely confusing or distorted from other viewpoints. For example, from one viewpoint a circle is drawn, yet from another viewpoint, it appears as a long thin oval. Snapshots allow a user to explicitly choose to draw in screen space and avoid this problem.

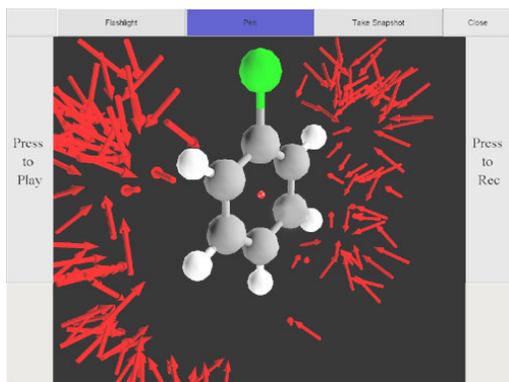


Figure 6. Difficult to draw on 3D molecule.

*Animated Transitions.* Releasing a snapshot has two effects. First, it places the snapshot in the 3D scene. Secondly, to preserve visual continuity, a smooth

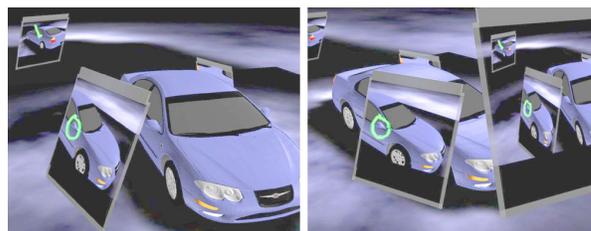
animated transition occurs between the original viewpoint in which the snapshot was taken and the current position of the display (see Figure 7). Depending on the movement of the display while the snapshot is frozen, it is possible to have the snapshot suddenly be positioned outside or behind the current viewpoint when unfrozen. Since the abrupt change of context can be very confusing for users, we employ animated transitions to prevent a visually jarring experience to keep users oriented.

Even when the viewpoint is unfrozen, marks can be made on the snapshots. This “always drawable” design minimizes the mode switching the user has to consider.



Figure 7. Animated transition after snapshot is released.

In keeping with the concept of omni-viewable widgets for a volumetric display [2], the snapshot frames are double-sided, in order to show the snapshot image and annotation ink on both the front and the back of the image plane (see Figure 8a). Thus, these annotations are viewable and writable from many vantage points.



(a) (b)

Figure 8. (a) Double-sided, viewable and writable snapshots. (b) Layering snapshots.

A convenient byproduct of spatially embedding snapshots is the ability to layer multiple snapshots (see Figure 8b). That is, users can take a snapshot of the 3D object with previously created snapshots in view. This allows users to capture snapshots within snapshots perhaps to annotate annotations within the shared 3D scene. We have noticed that there is a natural “onioning” effect in which the earliest snapshots are positioned closest to the center of the scene while new snapshots are positioned further away.

*Titlebars.* Populating the 3D scene with many snapshots can clutter and obscure the 3D object. To counteract this effect, we have added the ability to hide the snapshot images. Clicking on the titlebar of a snapshot toggles the visibility of the image. The titlebar is clickable from both the front and back sides, and is drawn semi-transparently to reduce the problem of obscuring the 3D objects. Lastly, while a snapshot is hidden, the position, size and orientation of the titlebar provide the user with an awareness as to the scope of the hidden annotation (see Figure 9).



Figure 9. Toggling snapshot visibility using titlebar.

*Recording.* To begin an annotation session, the user presses the Record button. This activates the capturing of all three data streams: view, voice and gesture information. The Record button turns red to indicate that data is being captured. Clicking on the button (see Figure 3f) again will end the annotation session.

*Playback.* To replay the last annotation session, the user can click on the Playback button located on the left side of the display (see Figure 3e). The annotation session is then played back in a separate window (see Figure 10). Note that during playback the 3D object and annotation data is also presented “live” in the main window. This real-time view of the annotation session allows the user to have alternative viewpoints during playback. This is useful, for example, if the recording is pointing out a flaw with a part of the model and the user wishes to get a closer look, or see the part from a different angle. Offering alternative 3D views has been done previously for navigation [31] but not for live annotations.

*Saving.* From the playback window, a user can choose to save the annotation session. A representative thumbnail image of the annotation session is generated and its use is described below.



Figure 10. Playback window.

*Workflow.* To begin a design review the user can select a 3D object from a 2D image browser (see Figure 11) running on the Boom Chameleon. Clicking on an image will launch our annotation software with the selected 3D model. Saved annotation sessions will also appear in the browser and if selected will playback the session. Users can transmit an annotation session to their colleagues by sending the data as an e-mail attachment.

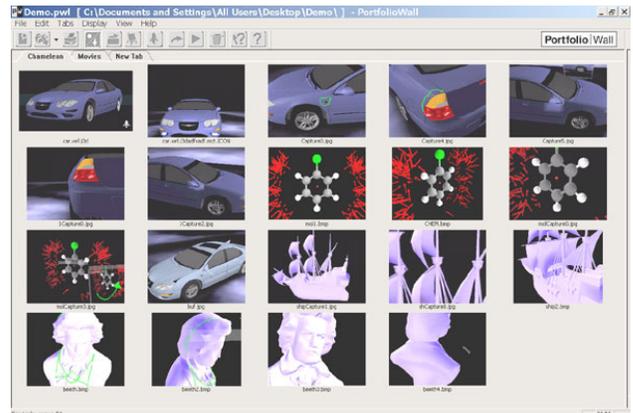


Figure 11. 2D PortfolioWall browser for accessing 3D models and annotations.

**IMPLEMENTATION**

The system uses a dual processor Pentium III 500 MHz computer with a NVIDIA GeForce2 graphics card. The annotation application is implemented in Java 3D. We use the VRML97 loaders provided by Sun to load VRML files into the scene.

The system is event based. A device driver synthesizes the various angles from the joints of the boom arm into a position and orientation value event for the display. Touch screen events are received as mouse events. The standard Windows sound API is used to record audio segments. A Java sound player is used to replay sounds.

Depending on the state of the interface, mouse events are synthesized into button presses and pointing gestures. Mouse drags are synthesized into a pen drawing events.

During recording, all events are time stamped and recorded. In addition, these events are used to update the display with the appropriate changes such as movement of the viewpoint or the creation of annotation marks in the scene.

During playback, the playback window is simply another view on the same 3D scene. Recorded events are sent to both windows but interpreted in different ways. In the playback window, all events are “replayed”; thus session viewpoint movements and annotations reappear in this window. In the main viewing window, recorded viewpoint events are ignored: the viewpoint in this view remains controlled by the boom position. Other recorded events such as flashlight pointing or drawing do occur in the main view. This allows a user to view the creation of annotations from viewpoints that differ from the viewpoint in the playback window.

Drawing on objects in the scene is accomplished by attaching Java3D behaviors to the scene geometry to detect the intersection of mouse events with scene geometry. When mouse drags occur, the system performs continuous picking on the scene geometry to determine the geometry intersection. This picking information is then used to place annotation marks that follow the scene geometry but are slightly offset above the geometry, to make them visible to the user.

Snapshots are created by first grabbing the image in the main window as a jpeg image. Next this image is used as a texture on both sides of a flat polygonal surface in the scene. These textures are placed in the 3D scene graph under one node for each snapshot. This allows snapshots to be quickly hidden or displayed by simply detaching or reattaching them to the scene graph.

## EVALUATION

The first version of the Boom Chameleon was built three years ago. This initial version was a “view only” device with no touch screen or microphone. This design limited users to move around a 3D scene and permitted no annotations. We have had extensive experience with hands-on demonstration of this initial version. It has been operated by hundreds of users at SIGGRAPH’99 as part of the trade floor exhibits. We have also conducted hundreds of hands-on demos in our laboratory with users ranging from experienced 3D computer graphics artists to “technology shy” executives of major automotive companies with absolutely no experience in operating 3D computer graphics camera controls. We have observed that 100% of the users immediately understand the navigation metaphor and are able to control their

navigation within seconds. We can only recall one user “getting lost” (a sculptor and expert user of a 3D computer graphics application) when she moved below the virtual floor and encountered an entirely black scene. We have also observed a few users initially confused when navigation started inside the car model. Because the interior of the car had not been modelled (you see only the back sides of the surfaces that make up the body) this presented an unfamiliar and confusing display. This situation was quickly remedied when the user randomly moved to the outside of the car.

Another interesting and often repeated observation is what we refer to as the “herding effect”. This effect occurs when small groups of people (up to 4) are trying to watch the screen of the Boom Chameleon while someone navigates. This results in a “follow the leader” situation where the group walks around with the user, looking over the user’s shoulder as the display is moved around. We have observed that this effect is especially pronounced when the Boom Chameleon user is the superior or manager of the others in the group. This method of following the user can at times be quite clumsy and ineffective so we have addressed the herding effect by providing a fixed duplicate display monitor close to the Boom Chameleon for others to watch. This concept suggests using the Boom Chameleon as a 3D podium where a presenter may drive the interaction while the audience watches on a larger fixed display.

We have also observed that some savvy 3D users realize the limitations of the viewpoint control technique. For example, some users asked how they could “fly above the car to look straight down from the top”.

With the Boom Chameleon augmented with a touch screen, microphone, and our annotation software, we performed a range of user evaluations. Approximately twenty-five people have tried and commented on the system with annotations. The comments from these sessions resulted in modifications to our annotation application. After these modifications, users had little trouble learning and operating the interface.

Our informal study was focused on how users simultaneously use the different types of annotations: speech, viewpoint control, drawing, and pointing. To study this behavior we recruited six subjects and asked them to perform a task with the Boom Chameleon. Before being given the task we explained to the subjects how the Chameleon and the various annotation facilities worked. During this training we had them inspect a 3D model of a car with no major defects. After this, we explained that their task involved searching for and reporting on some major defects that we introduced on the body of the car (see Figure 12). Subject were told to act as an insurance claim adjuster and that the car they were about to inspect

had been vandalised resulting in some severe and dramatically strange defects -- probably the work of an angry wizard. Subjects were also told that their task was to not only find the defects, but also to describe the defect and what needed to be corrected as if they were leaving a message for another person on how to correct the defect. To keep task times short, we added six very obvious defects to the car model (see Figure 12) ranging from a sagging driver's mirror, a crooked sunroof, an extra tail fin; an oddly rotated tire and some large dents in the body of the car.

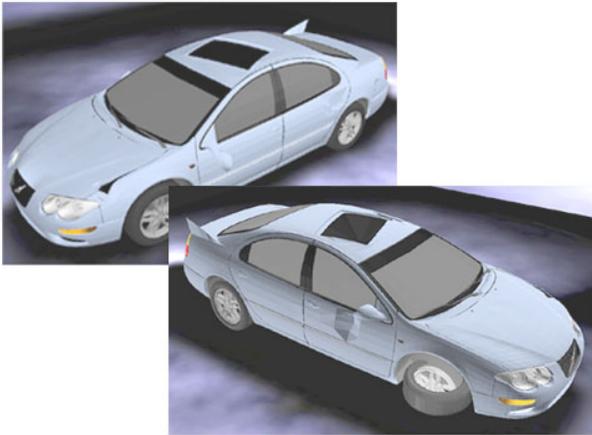


Figure 12. User study: 3D car model with defects.

We reviewed all of the annotation sessions and for two users, catalogued the annotation activity in detail. Figure 13 shows a visualization of the user's activities over time for their annotation session.

The first observation is that a high number of small annotation events are highly intermingled in this four minute time period. We believe that if annotation switching required explicit actions (e.g., switching between a viewpoint control mode and gesture input mode) a user's behavior would be quite different. During the annotation session, users spoke the majority of the time. This may have been due to slight nervousness about their performance in the task. Also, speech is at times occurring in parallel with viewpoint movement or gesture.

We observed that voice annotations were the predominant style of commenting in conjunction with "framing" the proper 3D view. The flashlight mode was used less frequently than the pen modes.

A variety of annotation session styles were observed. One user adopted a style in which they constantly were inspecting the model by repeatedly orbiting. Typically when a defect was spotted, they would leave their orbit and zoom in on the region of interest. For the most part, users would naturally segment the annotation session into a repeated sequence of actions per defect. For example, one user moved the viewpoint until she located each defect, shined the flashlight on the area of interest, then spoke and drew an annotation to comment on the defect.

We observed on a few occasions that the users would accidentally confuse drawing on object and drawing on screen modes. For example, users would make an annotation, move the viewpoint and then realize that they had not been drawing on the screen but drawing on objects in the 3D scene. Users seem to easily recover from this error by moving back to the appropriate viewpoint and either vocalizing a comment, or taking a snapshot and annotating it.

One surprising result was in how users employed the snapshot feature. While the test task could be performed easily without using snapshots, we observed that some users applied an annotation and then pulled back making a snapshot of it. One user in particular annotated all of the defects and then made a second pass taking a snapshot of each defect with its annotation. In both of these cases, we speculate that users took snapshots to emphasize the defects and to produce a visual summary.

### DISCUSSION & FUTURE WORK

Our experience with the Boom Chameleon and annotation software has revealed tradeoffs in determining the depth of placement for a snapshot in a scene. If a snapshot is placed very close to an object, the snapshot is large and close to the area of interest. However, this comes at the cost of the snapshot obscuring or even intersecting parts of the model. The snapshot does not obscure the model as

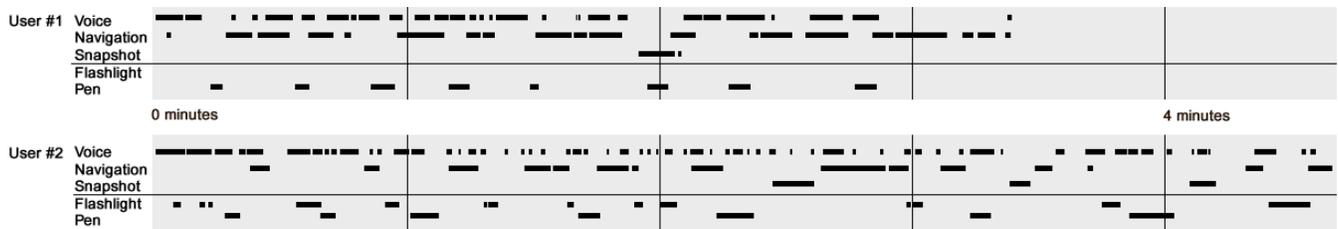


Figure 13. Annotation session for 2 users.

much, if the snapshot is positioned far away, however, the correspondence between model and snapshot is not as obvious. While turning snapshot on and off in a scene addresses these problems somewhat, future research may be to develop better schemes for users to control the depth or placement of snapshots.

Users were clearly adept at operating the Boom Chameleon and its annotation functions. However, what remains to be explored and developed are the conventions relating to the “new and best practices” for this method of annotation. For example, is there a common set of mark-up symbols that should be used? Should snapshots be used to organize annotations? Do you speak as if the recipient of the annotations is standing next to you?

The physical layout of the Boom Chameleon affords and limits certain types of navigation styles. As in our examples, using the boom to look at an object centered on the base of the Boom prevents the user from colliding with or interfering with the arm or base of the Boom. In contrast, using the unit to look from the inside of a virtual scene to the outside is difficult because there is only a small space to walk between the boom base and the display. Alternatively, if an object is centered at the center of the boom, the base acts as a physical barrier, not allowing the display to be moved through the center of the object. These problems could be alleviated by: a larger boom arm, inverting the boom to be mounted to the ceiling, or offsetting the origin of the virtual scene.

As the Boom Chameleon is a weighted and balanced device, it dampens user’s movement of the display and this results in a Steadicam™-like quality to the movements. We believe that it may be worthwhile to emulate this physical property in software for unsupported handheld spatially-aware displays.

Finally, future work could be to address many limitations of our system. For example, methods for deleting and editing portions of an annotation session may need to be introduced. Another avenue for future research is to explore the use of Chameleon navigation in combination with other types of camera controls, mappings and control/display ratios. For example, a graphical widget could be accessible on the screen to control zoom factor.

## CONCLUSIONS

Our work with Boom Chameleon has demonstrated that the “window on a virtual world” metaphor is an extremely quick to learn and easy to operate 3D navigation technique. While weakness of this approach is that it allows only a limited type of navigation and requires special hardware, we feel it nonetheless it could be used effectively in many applications where non-technical users require ability to inspect 3D virtual objects.

Our studies have shown that users utilized the simultaneous capture of 3D viewpoint, voice and gesture and easily transitioned between these data types. Because of this we believe that, with further refinements, the Boom Chameleon has the potential to be used effectively in 3D design reviews.

## ACKNOWLEDGMENTS

The authors would like to acknowledge the contributions of FakeSpace Laboratories, subjects who participated in our studies, Scott Guy, and Ravin Balakrishnan.

## REFERENCES

1. Art+Com System ([www.artcom.de](http://www.artcom.de))
2. Balakrishnan, R., Fitzmaurice, G.W. and Kurtenbach, G. (2001) User Interfaces for Volumetric Displays. *IEEE Computer*, 34(2), 37-45.
3. Bolt, R.A. Put-That-There: voice and gesture at the graphics interface. *SIGGRAPH Computer Graphics*, 14(3), 1980, 262-270.
4. Bourguignon, D., Cani, M-P. and Drettakis, G. (2001) Drawing for Illustration and Annotation in 3D, In *Proceedings of EUROGRAPHICS'01*, 114-122.
5. Buxton, W. & Fitzmaurice, G.W. (1998) HMD's, Caves & Chameleon: A Human-Centric Analysis of Interaction in Virtual Space, *Computer Graphics, The SIGGRAPH Quarterly*, 32(4), 69-74.
6. Chalfonte, B.L, Fish, R.S. and Kraut, R.E. (1991) Expressive Richness: A comparison of speech and text as media for revision, In *Proceedings of CHI'91*, 21-26.
7. Cohen, P., McGee, D., Oviatt, S., Wu, L., Clow, J., King, R., Julier, S. and Rosenblum, L. (1999) Multimodal Interaction for 2D and 3D Environments, *IEEE Computer Graphics and Applications*, 19(4), 10-13.
8. Fakespace Labs. ([www.fakespacelabs.com](http://www.fakespacelabs.com)).
9. Fitzmaurice, G.W. (1993) Situated Information Spaces and Spatially Aware Palmtop Computers. *Communications of the ACM*, 36(7), 38-49.
10. Fitzmaurice, G., Zhai, S. and Chignell, M. (1993) Virtual reality for palmtop computers, *ACM Transactions on Information Systems*, 11 (3), 197-218.
11. Frohlich, D. and Daly-Jones, O. (1995) Voicefax: A shared workspace for voicemail partners, In *Proceedings of CHI'95 Short papers*, 308-309
12. Hardock, G., Kurtenbach, G., and Buxton, W.A. (1993) Mate: A marking based interface for collaborative writing. In *Proceedings of UIST'93*, 259-266.
13. Harmon, R., Patterson, W., Ribarsky, W. and Bolter, J. (1996), The Virtual Annotation System, In

*Proceedings of Virtual Reality Annual International Symposium (VRAIS'96)*, 239-245.

14. Iris Annotator (SGI): ([www.sgi.com/software/annotator](http://www.sgi.com/software/annotator))
15. Jung T., Gross M. D. and Do E. Y., (2002), Annotation and Sketching on 3D Web Models, In *Proceedings of ACM Conference on Intelligent User Interfaces '02*, 95-102.
16. LaViola, J., Holden, A., Forsberg, D. and Bhuphaibool, D. (1998) Collaborative Conceptual Modeling Using the SKETCH Framework, In *Proceedings of the First IASTED International Conference on Computer Graphics and Imaging*, 154-158.
17. Levine, S. and Ehrlich, S. (1991) The Freestyle System: A Design Perspective. In Klinger, A. (Ed.) *Human-Machine Interactive Systems*, Plenum 3-21. Also in Baecker, R.M., Grudin, J., Buxton, W., and Greenberg, S. (Eds.) *Readings in Human-Computer Interaction: Toward the Year 2000*, 2nd edition, 871-880.
18. Lienhart, Rainer. (2000) A System for Effortless Content Annotation to Unfold the Semantics in Videos. In *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'00)*, 45-50.
19. Loughlin, M. and Hughes, J. (1994) An Annotation System for 3D Fluid Flow Visualization. In *Proc. IEEE Visualization '94*, 273-279.
20. Maeda, J., Greenwold, S. Installation. Aesthetics and Computation Group, MIT. <http://acg.media.mit.edu/people/simong/installationNew/cover.html>
21. Marshall, C.C. (1997) Annotation: from paper books to the digital library. *Proceedings of the Second ACM International Conference on Digital Libraries*, 131 – 140.
22. Moran, T. P., Chiu, P., Harrison, S., Kurtenbach, G., Minneman, S., and van Melle, W. Evolutionary engagement in an ongoing collaborative work process: a case study. *Proceedings of CSCW '96*. ACM, 150-159.
23. Morris, J.H., Neuwirth, C.M. et al. Virtual Work Room. School of Computer Science, Carnegie Mellon University. <http://muffin.hss.cmu.edu/vwr/web/overview.html>
24. O'Hara, K., Sellen, A., (1997) A comparison of reading paper and on-line documents. In *Proceedings of CHI '97*, 335-342.
25. Pedersen, E.R., McCall, K., Moran, T.P., Halasz, F.G., (1993) Tivoli: an electronic whiteboard for informal workgroup meetings. In *Proceedings of CHI '93*. 391-398.
26. Poupyrev, I., Tomokazu, N., and Weghorst, S. Virtual Notepad: Handwriting in Immersive VR, In *Proceedings of IEEE Virtual Reality Annual International Symposium'98 (VRAIS'98)*, 126-132.
27. Rekimoto, J. and Nagao, K. (1995) The World through the Computer: Computer Augmented Interaction with Real World Environments, In *Proceedings of UIST'95*, 29-36.
28. Schmalstieg, D., Encarnacao, L.M. and Szalavari, Z. (1999) Using transparent props for interaction with the virtual table," In *Proceedings SIGGRAPH Symposium on Interactive 3D Graphics 1999*, 147-154.
29. Small, D. and Ishii H. (1997) Design of Spatially Aware Graspable Display, *ACM CHI'97 Electronic Publications*.
30. Stellingwerff, Martijn. (1999) SketchBoX. In *Proceedings of Architectural Computing from Turing to 2000. eCAADe*, ISBN 0-9523687-5-7, Liverpool (UK) 15-17 September 1999, 491-497.
31. Stoakley, R., Conway, M. and Pausch, R. (1995) Virtual reality on a WIM: Interactive worlds in miniature. In *Proceedings of CHI '95*, 265-272.
32. Weber, K., and Poon, A. Marquee: A Tool for Real-Time Video Logging, *Proceedings of CHI '94*, 58-64.
33. Whittaker, S. Davis, R. Hirschber, J., and Muller, U. Jotmail: a voicemail interface that enables you to see what was said, *Proceedings of CHI'2000*, CHI Letters 2(1), 89-96.
34. Whittaker, S., Hyland, P., and Wiley, M. (1994) Filochat: Handwritten notes provide access to recorded conversations. *Proceedings of CHI '94*, 271-277.
35. Wilcox, L.D., Schilit, B.N., Sawhney, N. (1997) Dynamite: A Dynamically organized Ink and Audio Notebook. *Proceedings of CHI '97*, 186-193.
36. WindowsVR system [www.abs-tech.com/Produtos/3D\\_VR/VR-Hardware/hmds1/Virtual\\_Research/win\\_vr.html](http://www.abs-tech.com/Produtos/3D_VR/VR-Hardware/hmds1/Virtual_Research/win_vr.html).