

# Exploring and Reducing the Effects of Orientation on Text Readability in Volumetric Displays

Tovi Grossman, Daniel Wigdor, Ravin Balakrishnan

Department of Computer Science

University of Toronto

www.dgp.toronto.edu

{tovi, dwigdor, ravin}@dgp.toronto.edu

## ABSTRACT

Volumetric displays, which provide a 360° view of imagery illuminated in true 3D space, are a promising platform for interactive 3D applications. However, presenting text in volumetric displays can be a challenge, as the text may not be oriented towards the user. This is especially problematic with multiple viewers, as the text could, for example, appear forwards to one user, and backwards to another. In a first experiment we determined the effects of 3D rotations on text readability. Based on the results, we developed and evaluated a new technique which optimizes text orientation for multiple viewers. This technique provided 33% faster group reading times in a collaborative experimental task.

## Author Keywords

Volumetric displays, text orientation, reading speed.

## ACM Classification Keywords

H.5.3. [Information Interfaces And Presentation (e.g., HCI)]: Group and Organization Interfaces - *Collaborative computing*.

## INTRODUCTION

Volumetric displays [12], which project imagery in true three-dimensional (3D) space, present exciting new opportunities for interacting with 3D data. Users do not have to wear any supplementary hardware, and the 360° viewing angle allows multiple users to view the displayed imagery from anywhere around the display. These affordances, and recent empirical research showing benefits over existing 3D display technologies [15] potentially make volumetric displays well suited for collaborative 3D applications. However, perhaps because the technology is still fairly new, few interactive applications [16] have been developed for them. In particular, the intriguing scenario where multiple users collaboratively interact with the 3D volume (Figure 1) has yet to be explored.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2007, April 28–May 3, 2007, San Jose, California, USA.  
Copyright 2007 ACM 978-1-59593-593-9/07/0004...\$5.00.

Before developing applications for volumetric displays, it is desirable to first obtain a thorough understanding of some underlying issues associated with this new platform. In the present work, we concentrate on reading text in volumetric displays. While it is unlikely that volumetric displays will be used for extensive 2D text editing tasks, it is reasonable to expect that textual labels will be used. For example, one could imagine a team of architects viewing and labelling a marked-up 3D model of a new building. With this and other similar scenarios in mind, it is crucial that we be able to display text effectively on volumetric displays. A difficulty with presenting text in 3D environments is that it may not be oriented towards the viewer, resulting in suboptimal performance [5, 8]. This problem is intensified when within-world textual layouts are used, or when multiple users are viewing the 3D imagery from their own unique viewpoints.

In this paper, we first present an experiment aimed at quantifying the readability penalty that is incurred when text rendered in a 3D volumetric display is rotated – by a variety of angles about the pitch and yaw axes – such that it is not oriented facing the user. Based on the data from this experiment, we develop a technique for more optimally displaying text on volumetric displays. This technique is validated in a second experiment, where groups of three users simultaneously read text rendered on the volumetric display. Results show that our new technique decreases reading times for groups of three viewers by an average of 33%, making it a useful text rendering technique that could be adopted by designers of volumetric display applications.

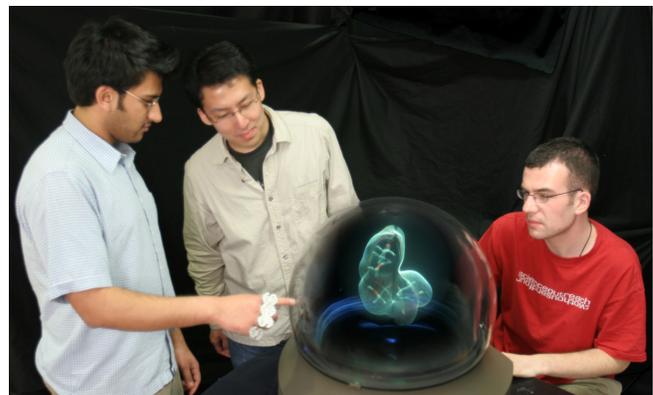


Figure 1. Multiple users can view and interact with imagery in a volumetric display from different viewpoints.

## Definitions

To aid our discussion of 3D rotation of text, it is useful to first define some terminology. Throughout this paper, we will refer to the rotation of text as *roll*, *pitch*, or *yaw*, with rotation angles specified in degrees. *Roll* is the rotation of text about the axis perpendicular to the face of the text, with a positive angle representing a counter clockwise rotation (Figure 2b). *Pitch* is the rotation of text about the axis that runs horizontal across the text's face, with a positive pitch meaning the top of the text is rotated forwards (Figure 2c). *Yaw* is the rotation of text about the axis that runs vertical across its face, with a positive yaw being a rotation with the left side of the text rotating forward (Figure 2d).



**Figure 2. Text rotations. (a) No rotation applied. (b) Positive roll. (c) Positive pitch. (d) Positive yaw.**

## RELATED WORK

There are three relevant areas of related work: interaction with volumetric displays, text orientation in traditional 2D environments, and text orientation in 3D environments.

### Interaction with Volumetric Displays

Although a number of systems have explored both single and multi-user interaction with shared 3D environments [1, 4, 6, 7, 9-11], very little work has been done to explore interaction issues unique to volumetric displays. Balakrishnan et al. [2] demonstrated interaction scenarios via wizard-of-oz prototypes. Although they did not explore text display per se, they did discuss strategies for displaying widgets for multiple viewpoints, such as rotating them towards the user, or displaying multiple copies for various viewpoints. A recent research prototype showed users directly interacting with volumetric displays using hand and finger gestures on and above the display surface [16]. While this work demonstrates a viable single user interaction scenario for volumetric displays, there has, to date, not been any exploration of the issues surrounding collaborative usage scenarios with volumetric displays.

### Displaying Text in 2D Environments

When text is rendered on a 2D display, roll is typically the only orientation variable of interest. Although there are some cases where a single user application may elect to render text rolled away from a traditional  $0^\circ$  orientation [14], it is primarily within the domain of co-located collaborative systems that orientation of on-screen objects is especially important.

When multiple users share a display, as in the present work, the orientation of information is slightly different for each user, as no two users are seated at precisely the same viewing angle to the display. In the most extreme cases, such as tabletop systems, text can appear upside down to

one user while facing another right side up. Wigdor and Balakrishnan [21] explore the use of orientation on tabletop systems, and experimentally evaluate how text orientation impacts reading speed on tabletop displays. Although informative, their work is limited in its application to 2D environments, since they limit their examination to the effects of roll. In the present work, we are concerned with the effects of text orientation in a shared, true 3D display environment. As such, an examination of the effect of pitch and yaw on reading speed is essential.

### Displaying Text in 3D Environments

Larson et al. [18] demonstrated some effects of 3D rotation of text on its readability. They found that reading speed was consistent with yaw of up to  $\pm 55^\circ$ , but that orientations beyond this had a significant effect. They also found that the effect was asymmetric: renderings where the first letter appeared smaller than the rest of the text tended to be read slightly slower than those where it was larger.

The present work varies from, and builds upon, Larson et al.'s work in several ways. First, Larson et al. examined the reading of text on a 2D perspective display. The volumetric display, in contrast, will render text in true 3D space, allowing viewers to change their viewpoint of the displayed imagery by moving their heads. Second, because they were concerned only with scenarios likely to occur on 2D displays, the rotations were limited to  $70^\circ$ . The present work demonstrates the effect of rotation about the full compass, necessary since users of volumetric displays can stand anywhere around the display. Third, their study was limited to examining the effect of rotation about the vertical axis (yaw), while the present work examines text rotated about both the vertical (yaw) and horizontal (pitch) axes. Finally, we will also explore the use of multiple reading aids, and develop a technique for optimizing text orientations for multiple viewers on a volumetric display.

Feiner et al. [13] define several methods for displaying abstract information related to a 3D world within windows on a heads-up display. Unfortunately, their designs are limited to a single user. Bell et al. [3] present an algorithm that dynamically improves the positional layout of text within a 3D environment; however, their solution is also limited to a single user. Bowman et al. [5] and Chen et al. [8] implemented two methods for rendering text in a 3D scene. With a heads-up display metaphor, text is rendered within the view plane, and thus is never pitched or yawed. In their within-world display metaphor, text is projected onto faces of objects in a 3D scene, thus undergoing pitch, yaw, and roll transformations. An alternative they discuss but do not implement is displaying text within-world but to dynamically rotate it to always face the user. Their experiments found the static within-world display to be detrimental to reading, supporting the hypothesis that the 3D rotation of text will increase reading time. Precisely quantifying the effects of these rotations is the goal of our first experiment.

### TEXT ROTATION AND DISAMBIGUATION

In volumetric display applications, when text is displayed using a within-world layout, or when multiple viewers are present, text may not be oriented towards a user. Under most usage scenarios, it is unreasonable to expect a user to walk around the display just to read a textual label. While one alternative is to present textual labels using a heads-up display metaphor, this would require the designer to clutter up the display space with multiple labels for each individual user. Thus, it is valuable to consider the scenario where a user is required to read text under a 3D rotation. An impediment to recognizing words under such a condition is the introduction of visual ambiguities. As we see in Figure 3a, rotational transformations of the text result in ambiguity to not only the word, but also the individual characters.

While surrounding asymmetric letters could disambiguate rotated text, our preliminary observations showed that the ambiguities can still cause confusion and frustration for the reader. To address this, we experimented with the following techniques to allow the reader to correctly interpret the text.

#### Underline Disambiguation Technique

In this technique, we employ the relatively lightweight strategy of underlining the first character of the word (Figure 3b). This provides disambiguation by identifying both the first character and the bottom of the word.

#### Uppercase Disambiguation Technique

While the underline disambiguation technique indicates the rotation of the entire word, we also wished to seek out a strategy which could disambiguate individual characters. To do so, we required a typeface in which characters were not ambiguous under rotations. Upper case letters were chosen as a suitable candidate (Figure 3c), for which the only possibly ambiguity would be ‘M’ and ‘W’. While displaying text in uppercase can reduce reading speeds [20], we expect the effect to be minimal when reading short textual labels. Furthermore, this technique for reducing character ambiguities is simple and easily implemented. If successful, more subtle approaches to reducing character ambiguities could be developed, such as custom fonts.

- (a) bob pop dod  
 (b) bob pop dod  
 (c) BOB BOB BOB

Figure 3. The word “bob” under no rotation (left), 180° pitch (middle), and 180° yaw (right). (a) No disambiguation. (b) Underline disambiguation. (c) Uppercase disambiguation.

### EXPERIMENT 1

In this experiment, we seek an understanding of how text orientation in a 3D environment will affect reading times. We wish to quantify the relative effect of a range of rotations and identify particularly problematic ones. We also wish to measure the efficacy of our two disambiguation techniques.

This experiment will be similar in design to the study performed by Wigdor and Balakrishnan [21], which tested the effect of text orientation on tabletop displays. However, since the text on such displays is inherently 2D, only the effect of roll was examined. In this experiment, we will evaluate the effect of pitch and yaw on reading times.

We can easily predict some results, such as higher reading times when text is upside down or backwards, as well as a significant spike in the reading times at orientations in which the text is parallel to the viewer’s line of sight. Less predictable, however, is at what point, how quickly, and how significantly reading times will increase to these maximal peaks. As for our two disambiguation techniques, we expect that both techniques will decrease reading times in some cases, but we wish to determine under which orientations these techniques are most beneficial, and what the relative efficacy of the two techniques are.

### Apparatus

We used a 3D volumetric display developed by Actuality Systems ([www.actuality-systems.com](http://www.actuality-systems.com)), which generates a 10” spherical 3D volumetric image by sweeping a semi-transparent 2D image plane around the Z (vertical) axis. A total of 198 2D images (slices) of 768x768 pixels each are displayed uniformly around the Z-axis. The display’s refresh rate is 24Hz. Words were rendered using the default OpenGL Utility Toolkit mono-spaced font. A standard QWERTY keyboard, which was placed directly in front of the participant, was used for text entry. To maximize the image quality of the volumetric display, the room lights were turned off, and a small light was used to illuminate the keyboard. Participants sat in a fixed location chair whose height was adjusted so that all participants viewed the imagery from the same viewpoint regardless of the individual’s height. We used a Vicon motion tracking system ([www.vicon.com](http://www.vicon.com)) to track the positions of the viewers’ heads. Three passive reflective markers were placed on a hat which participants wore. Six Vicon cameras tracked the 3D location of these markers, and the data was streamed into our experiment application at 120Hz. The application ran on a Pentium 4 PC running at 2 GHz, with imperceptible latency. Figure 4 illustrates the apparatus.



Figure 4. Hardware apparatus and setup for Experiment 1.

## Participants

Twelve unpaid volunteers (3 female, 9 male), ranging in age from 20 to 35, participated in the experiment. All participants read English natively, and had proficient typing skills.

## Procedure

To maximize the general applicability of our results, we abstracted the primary task of interest, reading and comprehending text in a volumetric display, and conducted a basic speed-of-reading test. Participants were presented with a series of common five-letter words which they were asked to read and type into the system. This design was similar to that described in Wigdor and Balakrishnan [21].

To begin a trial, participants pressed “enter”, and the word was presented. We wished to measure the time participants took to read the word before beginning to type. They were instructed to press “enter” as soon as they recognised the word, which would then disappear, ensuring that they were no longer reading the text. A text box was then displayed, into which they would type the word and then press “enter” to finish the trial. If they typed an incorrect word, a tone would sound and the trial word would appear again, with the trial counted as an error. If, after beginning to enter their response, the participant wished to review the word, they could press “escape” to return to viewing the word – all such trials were also counted as errors. Lastly, if the participant failed to enter the correct word after 30 seconds, the trial would end and be counted as an error.

Because volumetric displays are still in their infancy, their quality is relatively low in comparison to 2D displays. Grossman and Balakrishnan [15] provide a detailed description of how certain locations are difficult to see. To ensure this did not affect reading times, we chose a stable and visible location to display the word throughout the experiment. The word was rotated with either a pitch or a yaw. To keep the size of the experiment manageable, we did not vary both pitch and yaw in the same condition.

At the start of the experiment, a short calibration program was used to control participants’ viewpoints. This was done by adjusting the chair height, and using the motion tracking system to ensure that viewpoints were aligned with the predefined location of the text in the volumetric display. Furthermore, to start any trial, the participant’s viewpoint first had to be horizontally and vertically aligned with the center of the text location, with a 1° error allowance. Once the trial began, users were free to move their heads and stand up, as long as their feet did not move.

## Design

A repeated measures within-participant design was used. The independent variables were the disambiguation technique, *DT*, the rotation type, *RT*, and the rotation angle,  $\theta$ . The values for *DT* were *none*, *underline*, *uppercase*, and *both* (both underline and uppercase). The rotation type was either *pitch* or *yaw*. The angle of either the pitch or yaw,  $\theta$ , took on the 24 values between -180° to 165° in increments of 15°, resulting in 48 text orientations.

The experiment lasted about 90 minutes, and was divided into 4 sessions. Each session consisted of all trials for 1 of the 4 values of *DT*. Each session was divided into 3 blocks, within which the 48 text orientations appeared 2 times each in random order. This resulted in 1152 trials per participant. The five-letter words were chosen from among the most frequent five-letter words appearing in the British National Corpus, with the relative frequency of the set of words chosen for each condition balanced within each participant.

To familiarize participants with the task and disambiguation techniques, four warm-up trials were given. Presentation orders of the disambiguation techniques were counterbalanced using a 4x4 balanced Latin square design. Participants were randomly divided into one of four groups, with each group being assigned one of the four orderings.

## Results

### Reading Time

Reading time is the primary dependent measure, defined as the time between the word appearing and the user pressing “enter” before beginning typing. Recall that once users did this, the text disappeared, ensuring that users had finished reading at this point. In our analysis of reading time, we discarded trials in which errors occurred (5.1% of the data), and removed outliers that were more than 3 standard deviations from the group mean (2.26% of the data).

Repeated measures ANOVA showed main effects for *DT* ( $F_{3,33} = 224, p < .0001$ ), *RT* ( $F_{1,47} = 581, p < .0001$ ), and  $\theta$  ( $F_{23,1081} = 238, p < .0001$ ), with significant interactions between all pairs (all  $p < .0001$ ). Text with pitch rotations were more difficult to read, with reading times of 1.96s for  $RT = \textit{pitch}$ , and 1.23s for  $RT = \textit{yaw}$ .

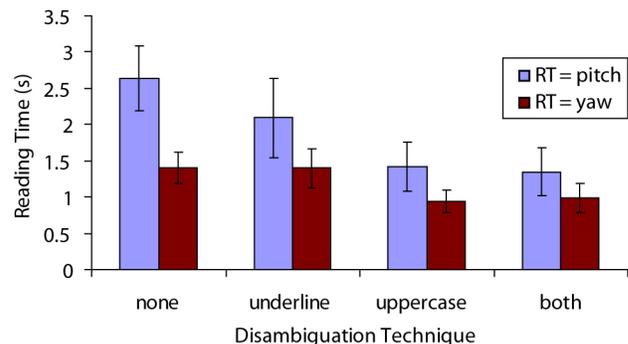


Figure 5. Reading times for each disambiguation technique.

*DT* had average times of 2.06 for *none*, 1.79 for *underline*, 1.20 for *uppercase*, and 1.18 for *both*. Post hoc multiple means comparison tests showed that *none*, *underline*, and *uppercase* were all significantly different ( $p < .05$ ), while *uppercase* and *both* were not. The same significant effects were seen for the conditions where  $RT = \textit{pitch}$ , however for the conditions where  $RT = \textit{yaw}$ , there was no significant difference between *none* and *underline*, or between *both* and *uppercase* (Figure 5). In other words, the effect of the underline disambiguation technique was only significant when the text was pitched, and had no significant effect when the words were already being presented in uppercase.

Figure 6 shows the reading times for  $RT = pitch$ . Although reading times seem to increase at pitches of  $60^\circ$  and  $-60^\circ$ , post hoc analysis shows that for all values of  $DT$ , none of the reading times within the  $[-60^\circ, 60^\circ]$  interval are significantly different (all  $p < .05$ ), showing that pitching the text within this interval has negligible effects. Once outside this interval, the data clearly spikes when the text is close to being parallel with the user's line of sight ( $\theta = \pm 90^\circ$ ). Past this point, reading times drop back down, but since the text remains upside down, reading times are slower than they were within the  $[-60^\circ, 60^\circ]$  interval.

In comparison to the condition  $DT = none$ ,  $DT = underline$  only reduced reading times for  $\theta = -165^\circ, -135^\circ, -120^\circ, 120^\circ, 150^\circ, 165^\circ$  and  $180^\circ$ . However, the condition  $DT = uppercase$  reduced reading times for all  $\theta \geq 90^\circ$  and all  $\theta \leq -105^\circ$ .  $DT = both$  produced reading times which were lower than  $DT = uppercase$  outside the interval  $[-120^\circ, 120^\circ]$ , but this difference was not significant.

Interestingly, this graph is not completely symmetrical. Notably, with  $DT = none$ , pitches of  $90^\circ$  and  $105^\circ$  caused higher reading times than pitches of  $-90^\circ$  and  $-105^\circ$ . This is likely because it is easier for users to raise than to lower their viewpoint when seated. With  $-90^\circ$  and  $-105^\circ$  pitches, raising the viewpoint results in seeing the text right-side-up and forwards, while for pitches of  $90^\circ$  and  $105^\circ$ , raising the viewpoint results in backwards and upside-down text.

The reading times for  $RT = yaw$  are shown in Figure 7. Post hoc analysis shows that no data points are significantly different when  $\theta$  is within the interval of  $[-75^\circ, 75^\circ]$ . Note that this interval is slightly larger than the interval for which pitch was not significant. As with the yaw rotations, the reading times spike for  $\theta = -90^\circ$  and  $90^\circ$ , where the text is parallel to the user's line of sight, and physical movement is required in order to see the text at all. Past this point, the data drops back down but remains slower than within the  $[-75^\circ, 75^\circ]$  interval as the text appears backwards.

As for the disambiguation techniques, *underline* only significantly reduced times at  $\theta = 135^\circ$ . *Uppercase* reduced times significantly for all values of  $\theta$  outside the interval  $[-90^\circ, 90^\circ]$ , except for  $-150^\circ$ . As with the pitch data,  $DT = both$  provided no further advantage over  $DT = uppercase$ .

Unlike the pitch data, which showed some positive-negative asymmetry in the significant differences, the yaw data is completely symmetric with respect to statistical significance, showing that a yaw rotation has equivalent effects in either direction. We do not claim that this result contradicts earlier results [18] since our participants could move their heads to improve their viewpoints. We believe that, unlike with *pitch*, the effect was symmetrical because users could just as easily move their heads left as right.

#### Head Movement

We also measured participant head movement while they were reading text, defined as the length of the path the head took during the reading phase of the trial. The effect of text

orientation on head movement is similar to the effects on reading time. Head movement was significantly affected by RT ( $F_{1,47} = 218, p < .0001$ ), averaging 6.95 inches for RT = yaw and 11.4 inches for RT = pitch.

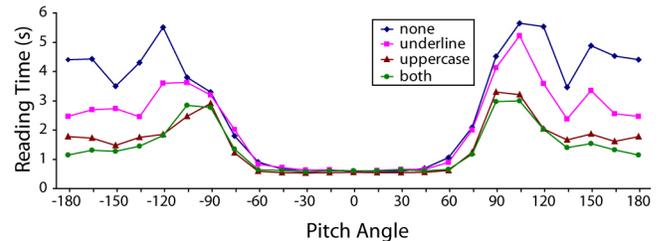


Figure 6. Reading times by pitch rotation angle. In Figures 6 and 7 the data for angle =  $-180^\circ$  is repeated at angle =  $180^\circ$ .

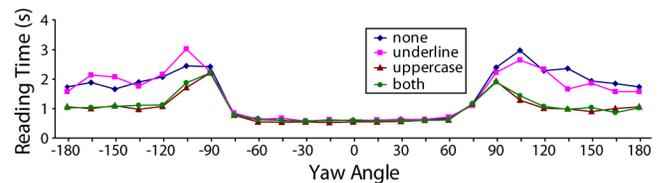


Figure 7. Reading times by yaw rotation angle.

#### ORIENTATION OPTIMIZATION

Our results from Experiment 1 show that even with an unambiguous font, there are still readability problems for text which is rotated in 3D space. Although within-world layouts – where text is rendered on faces of objects in the 3D space – have desirable properties, these results tell us that designers should not use a naïve within-world layout for displaying text within the volumetric display. We now discuss an orientation optimization strategy, which maintains the beneficial properties of within-world layouts, while optimizing reading times for multiple viewers.

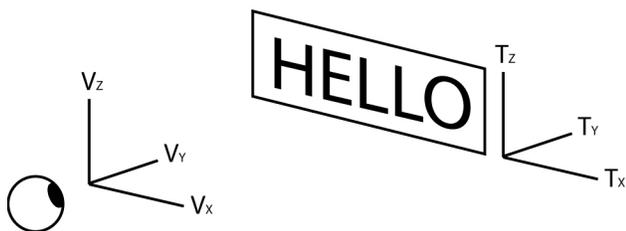
The goal for the technique is to present text at the best possible orientation for all users, while still allowing some rotation for within-world display of information. The algorithm searches through possible text orientations, and for each one, estimates reading times based on the orientation of the text relative to each user's viewpoint. The algorithm then chooses the text orientation which minimizes the average estimated reading times across all viewers. It is important to note that this technique does not require our estimated reading times to be precise. Our algorithm only requires the estimates to be positively correlated to actual reading times.

A necessary tool for this technique is a function which, given the position of a user's eye and the relative orientation of text, would provide an estimate of the time required for that user to read the text. To begin, we define  $Pitch(\theta)$ ,  $Yaw(\theta)$ , and  $Roll(\theta)$ , which yield reading times for their respective rotation type for an angle of  $\theta$ . The return values for the  $Pitch$  and  $Yaw$  functions are calculated by interpolating over the data points obtained in Experiment 1 (Figure 6 and Figure 7). Return values for the  $Roll$  function are calculated by interpolating over the data points reported in Wigdor and Balakrishnan [21].

If the text's orientation relative to that user's line of sight contained only one of these three rotation types, we could simply use the corresponding function and we would be done. However, the reading time estimation is more complex, because we now need to consider the case where multiple such rotations exist. For example, if text had 180° rotations of both pitch and roll, the text would not be upside down. It would be incorrect to use the data we obtained in Experiment 1 for pitch = 180°, because those reading times were based on upside down text. Our method is to define viewer axes ( $V_X$ ,  $V_Y$ ,  $V_Z$ ), and text axes ( $T_X$ ,  $T_Y$ ,  $T_Z$ ). The viewer axes are defined as the vector through which the user is looking ( $V_Y$ ), the up vector perpendicular to this ( $V_Z$ ), and the cross product of these two vectors ( $V_X$ ). The text axes are defined as the vector in which the text runs ( $T_X$ ), the normal vector of the plane on which the text is placed ( $T_Y$ ), and the cross product of these two vectors ( $T_Z$ ) (Figure 8). If the text is perfectly oriented to the user, then these three axes would all be the same, so we consider the divergence between these three pairs of axes. For example, consider  $T_X$  and  $V_X$ , with an angle between them of  $\theta$ . The angle  $\theta$  is due to a combination of only yaw and roll rotations, as a pitch rotation would not affect  $V_X$  or  $T_X$ . To estimate the amount of  $\theta$  which is due to yaw rotation we take the length of the projection of  $T_X$  onto  $V_Y$  ( $L_1$ ) and for Roll we take the length of the projection of  $T_X$  onto  $V_Z$  ( $L_2$ ). We then take the following weighted average of our reading time functions,  $Yaw(\theta)$  and  $Roll(\theta)$ :

$$Time = \frac{Yaw(\theta)L_1 + Roll(\theta)L_2}{L_1 + L_2}$$

We then repeat this calculation for the  $T_Y - V_Y$  divergence and  $T_Z - V_Z$  divergence. Our final estimation is the average of these three calculated reading times. By taking this average, the reading times for some orientations may be underestimated. Despite this, we expect the estimates to correlate with actual reading times, and therefore serve its purpose in the algorithm.



**Figure 8.** Viewer axes ( $V_X$ ,  $V_Y$ ,  $V_Z$ ) and text axes ( $T_X$ ,  $T_Y$ ,  $T_Z$ ) definitions used by the orientation optimization algorithm.

We determine the optimal orientation by searching through all combinations of pitch, yaw, and roll of the text to find the minimum estimated average reading time across all viewers. To allow for consideration of within-world layouts, the algorithm takes an additional parameter, which is the maximum allowed divergence in degrees from the original orientation in any of these axes of rotation. This allows designers to minimize the amount of display volume real estate consumed, and allows users to continue to

associate text with the corresponding aspects of the scene. However, even with the maximum divergence defined, we still allow our algorithm to flip the text by 180° along any axis, since the text will continue to be rendered on the same plane, retaining its essential “within world” quality. The running time of this brute force algorithm is thus based on the granularity of the search and the maximum allowed divergence. With sufficiently coarse granularity of the search, this approach runs in real time.

## EXPERIMENT 2

To evaluate the effectiveness of our orientation optimization technique, we conducted a second experiment. Along with testing this new technique, we will make two adjustments to the experimental procedure.

First, since we are testing a technique to aid reading times when multiple viewers are present, we will perform the study with groups of three participants, all reading the same word at the same time. Previous work [16] has claimed that a volumetric display's 360° viewing angle provide the potential for collaborative use. This study provides an opportunity to validate this claim, with the implementation of an actual collaborative usage scenario. This differs from our own Experiment 1, along with the previous work evaluating text orientations on tabletop displays [21], which were both conducted with a single participant at a time.

Second, we will run the experiment in a more realistic usage scenario, by projecting the words onto faces of a cube. This is an abstraction of a task in which multiple users are viewing a 3D model which is labelled with textual information, using a within-world layout. In such a scenario, the textual labels will rotate in 3D space with the model. As we know from Experiment 1, this will cause reading difficulties at some orientations, justifying the need for our orientation optimization technique.

## Apparatus

The same apparatus from Experiment 1 was used. The three users were seated at 120° from each other around the display. Three keyboards were used, one in front of each user, which allowed for text input by all users concurrently. A small flashlight was positioned above each of the keyboards to illuminate them. Figure 9 illustrates the setup.



**Figure 9.** Hardware apparatus and setup for Experiment 2.

### Participants

A total of 12 new unpaid volunteers (11 male, 1 female), consisting of 4 groups of 3, ranging in age from 19 to 32, participated in the experiment. All participants were fluent in English and had proficient typing skills.

### Procedure

The procedure was similar to that of Experiment 1, with the primary difference being that three participants completed the experiment simultaneously. To begin a trial each user pressed “enter”. Again, we measured the amount of time it took users to read a five-letter English word before typing it into the system. Once the user had read the word, they would press “enter” again.

Unlike in the first experiment, the word did not disappear at this point, since the other two participants may have still been reading it. Instead, a dense grid was displayed between the participant and the word to indicate that they should no longer read the text. Because it is impossible to render opaque imagery on the current generation of volumetric displays, it was still possible for the participant to “cheat” and read the text. However, participants were under observation, ensuring that they would not do so. When participants pressed “enter”, a text box appeared for them to enter the word. To ensure that participants were not reading from one another’s text boxes, answers were “password hidden”, with characters rendered as “\*”. Once finished typing, they pressed “enter” again to submit their response. If the input was incorrect, an audio cue would prompt the participant to try again. The trial ended when all three participants submitted the correct response. If a participant had not finished the trial after 20 seconds, the trial would end, being counted as an error for that participant.

Users were seated in three height adjustable chairs directly in front of the volumetric display. The height of each participant’s viewpoint was set to be equal with the center of the volumetric display. Once a trial began, users were allowed the same head movements as in Experiment 1.

A wireframe cube was rendered in the center of the volumetric display. The cube took on various pitch and yaw rotations, and for each trial, one face of the cube was labelled with text. For each cube orientation, there were six trials, one for each face of the cube. Words were displayed in uppercase, as this was the most effective technique in Experiment 1. The underline cue was not used as it provided no effect when words were already uppercase.

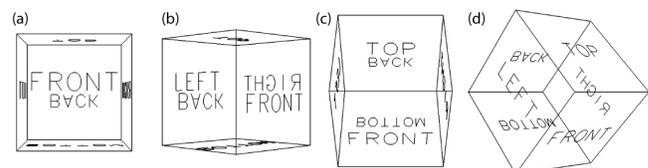
The layout of the text was either a naïve within-world projection onto the cube face, or optimized using our orientation optimization algorithm. Although we were capturing the head positions of each user in real time, the three viewpoints used in the algorithm were constant throughout the experiment, based on the location of their chairs, with the height set to the centre of the volumetric display. This allowed us to pre-compute the optimal orientation to a high degree of precision (with a granularity of 1°) using our orientation optimization technique. This

simplification also allows us to determine if our algorithm can be effective when precise viewpoint locations are not known. A maximum divergence of 15° was used for the algorithm, which would give it enough freedom to avoid the major spikes seen in Figure 6 Figure 7. The algorithm used the data from Experiment 1 for  $DT = \text{uppercase}$ , since all words were displayed in uppercase for this experiment.

Unlike Experiment 1, it was impossible to design an acceptable experiment in which all three participants were always viewing the text in an ideal area of the volumetric display. As such, we expected to see higher than expected reading times for some trials. While this would have been inappropriate for our first experiment, it was acceptable for the purposes of this experiment, since the goal was no longer to determine average reading times for various orientations, but to test our orientation optimization technique in a real usage scenario.

### Design

A repeated measures within-participant design was used. The independent variables were *LAYOUT* (naïve, optimized), cube pitch, *PITCH* (0°, 45°), cube yaw, *YAW* (0°, 45°), cube face, *FACE* (left, right, top, bottom, front, back), and target user, *USER*, (1, 2, 3). The naïve layout of the text on each face was determined in a manner such that we obtained a good range of text orientations throughout the experiment. Figure 10 shows the naïve text layout and the four possible orientations which the cube took on. The labels in the *optimized* condition diverged from the illustrated naïve layout by a maximum of 15°, as determined by our optimization algorithm. The *USER* variable indicates which user the cube orientation was relative to. When *USER* = 1, the cube orientations seen in Figure 10 were seen by viewer 1. By dint of their seating positions, for *USER* = 2, the cube yaw was increased by 120°, and when *USER* = 3, the cube yaw was increased by 240°. This design ensured that by the end of the experiment, all users had seen exactly the same set of cube and text orientations. This design resulted in 12 possible cube orientations (2 pitch x 2 yaw x 3 participants).



**Figure 10. Experiment cube orientations, with the naïve text layout. (a)  $pitch = 0^\circ$ ,  $yaw = 0^\circ$ . (b)  $pitch = 0^\circ$ ,  $yaw = 45^\circ$ . (c)  $pitch = 45^\circ$ ,  $yaw = 0^\circ$ . (d)  $pitch = 45^\circ$ ,  $yaw = 45^\circ$ . These four rotations were repeated relative to each of the three viewers.**

The experiment lasted approximately 70 minutes, and was divided into two sessions. Each session consisted of all trials for one of the two values of *LAYOUT*. In each session the 12 possible cube orientations appeared 4 times in random order. For each cube orientation, 6 trials were completed, one for each face of the cube, also in random

order. This design resulted in 576 trials per subject. The same five-letter words which were used in Experiment 1 were used in this experiment, with each group receiving the words in a different order. As in the first experiment, the words were distributed between the 2 sessions such that the average frequency of the words within each session had a similar average frequency in the British National Corpus.

To familiarize participants with the task, six warm-up trials were given before the session began. Ordering of the layouts was counterbalanced across the four groups.

## Results

### Individual Reading Times

We again define reading time as the time to press “enter” before entering the text. Because text did not disappear in this experiment, we removed outliers of typing time (0.9% of data). We also discarded trials in which errors occurred (5.7% of data), and reading time outliers (1.9% of data).

Repeated measures ANOVA showed a main effect for *PITCH* ( $F_{1,23} = 199$ ,  $p < .0001$ ), *LAYOUT* ( $F_{1,11} = 465$ ,  $p < .0001$ ), and *FACE* ( $F_{5,115} = 142$ ,  $p < .0001$ ), but not for *YAW* or *USER*. There was no effect for *USER* since this variable only indicated to which user the cube orientation was relative, so for each value of *USER* the exact same text orientations were seen. There was no significant effect for *YAW* since this variable only defined the cube orientation relative to the target user, and not the orientation of the text.

The most important effect seen here is that the layout had a strong and significant effect. With the naïve layout, reading times averaged 2.13s, and with the optimized layout, reading times averaged 1.42s, a 33% improvement. The layout also showed significant interaction with *FACE* ( $F_{5,55} = 39.8$ ,  $p < .0001$ ), *PITCH* ( $F_{1,11} = 6.31$ ,  $p < .05$ ), and *YAW* ( $F_{1,11} = 17.7$ ,  $p < .0001$ ), lessening the effects of each.

If we combine the variables *PITCH*, *YAW*, *FACE*, and *USER*, we get 72 unique text orientations which each participant saw throughout the experiment. When we look at each of these 72 orientations, in 55 of them the average reading time was lower with the optimized layout. Of the remaining 17, only 3 orientations had significantly slower reading times ( $p < .05$ ). Before concluding that our algorithm failed in these 3 of 72 conditions, we must recall how it worked. The algorithm was designed to optimize the *average* reading time for all viewers for any given trial. So in some cases, the algorithm may increase an *individual's* reading time, in order to reduce the other two viewers' reading times, to obtain the best average reading time for the *group*. This was indeed the case for the orientations where the optimized layout was significantly slower.

### Group Reading Time

Another variable in which we were interested was the average group reading time for each trial. We wished to know if our algorithm successfully reduced this time, since that is what it was designed to do. We numbered the orientations from 1 to 24, for the 24 possible text

orientations which the group saw (2 *YAW* x 2 *PITCH* x 6 *FACE*). Each of these orientations were repeated for the 3 values of *USER*. As with individual reading times, *USER* did not significantly effect average reading times, as this only signified which user the orientation was relative to.

Repeated measures of analysis showed a main effect for both the orientation number ( $F_{23,161} = 73.8$ ,  $p < .0001$ ) and the layout ( $F_{1,3} = 499.9$ ,  $p < .0001$ ), with a significant interaction between the two variables ( $F_{23,69} = 14.7$ ,  $p < .0001$ ). Figure 11 shows the average reading times for each of the 24 possible orientations. The reading times were faster for the optimized layout, in all of the 24 conditions, and significantly so in 10 of the conditions ( $p < .05$ ).

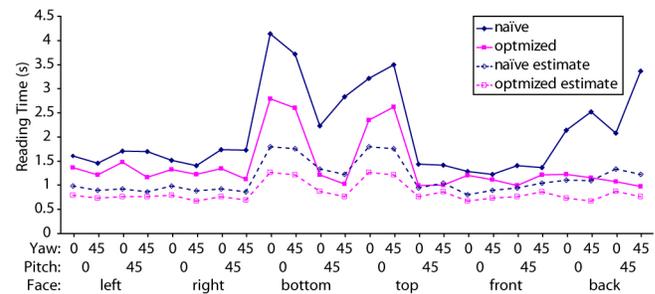


Figure 11. Actual and estimated group average reading times for each layout and orientation.

This result shows that our optimization algorithm performed as intended, reducing average reading times over the entire group. Figure 11 also includes the estimated average reading times which our algorithm used to optimize the text orientation. By definition, the estimated times with the optimized layout are always lower than that of the naïve layout. As expected, the algorithm underestimated the reading times, however, there is still a visible correlation between the estimated and observed times. Regression analysis gives an  $r^2$  value of 0.85, with  $a$  and  $b$  values of 2.45 and -0.67 respectively (Figure 12).

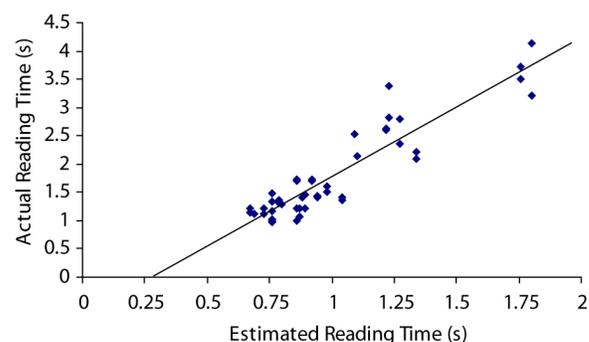


Figure 12. Comparison of estimated and actual reading times.

### Head Movement

As in Experiment 1, we measured the participant's head movement while they were reading text, defined as the length of the path which the head took during the reading phase of the trial. Head movements were significantly affected by the text layout ( $F_{1,11} = 320$ ,  $p < .0001$ ), with averages of 14.65 inches for the naïve layout, and 8.15

inches for the optimized layout. This is an important result, as the optimization algorithm reduced the need for head movements. The head movements seen in this experiment were also higher than what was observed in Experiment 1. This explains the higher reading times of similar orientations between the experiments, as the poor quality forced users to move their head to obtain a better viewpoint.

## DISCUSSION AND CONCLUSION

We have presented two experiments exploring the effects of 3D rotations on text readability in true 3D volumetric displays. Our motivation was a scenario in which the user is required to read text that may not be oriented towards them. This can occur either with the presence of multiple users, or with a desire to maintain a within-world layout. Additionally, we discussed potential issues which could increase reading times, and suggested two disambiguation techniques to aid reading, both of which were shown to be beneficial in an initial experiment. Based on the results of that experiment, we developed a new technique to optimize text orientation for multiple viewers. In Experiment 2, we validated this new technique within a collaborative experimental task.

Our findings are consistent with, and extend, the literature. First, we found that text not oriented towards the user can increase reading times [5, 8]. Similar to the findings in Larson et al. [18], we found that yaw only began to have an impact when the text was rotated by 75°. In contrast to this previous work, we found that in an environment allowing head movement, the effect of yaw on reading speed was symmetrical. We believe that this is because, with head movement, users reduce the distortion of the text, and so the effect of first-letter dominance is mitigated.

One of the main contributions of this paper is the orientation optimization algorithm which we proposed and validated. The algorithm optimizes orientations for any number of viewers, regardless of their relative viewpoints. We conducted an experiment and found that it reduced reading times for 3 viewers by 33%, despite a number of simplifications. Firstly, although the effects of pitch, yaw and roll are likely dependent, we used only independent data for the effects of these rotations in the design of our algorithm. Even so, our algorithm was successful, and produced estimated reading times which had a positive correlation to actual reading times. Another simplification was that the algorithm estimated the viewpoints of the users based on their seating positions. Although using real time head position data may have further improved results, we have shown that our algorithm can be implemented without the need for head tracking technology.

Another advantage of the optimization algorithm was that it significantly reduced participants' head movements. This could be especially important for scenarios in which head movements are inadequate for reducing the orientation of text. For example, if the volumetric display were significantly larger, head movements would not change the

viewing angle as significantly for some text. Similarly, in a virtual reality environment, it is much more difficult to navigate to a suitable viewing position [5].

It was interesting to observe in Experiment 2 that the head movements of the users did not interfere with each other. The 120° spacing was enough for the users to move freely as required. However, if there were more users, or the users were closer together, it would again become critical for required head movements to be limited, as the users' head movements could physically interfere with one another.

Another contribution of our work was the first implementation of collaborative interaction on a volumetric display. Even though the interface and task were very simple, there were some interesting issues we encountered which will be relevant to future groupware applications developed for volumetric displays. Firstly, because all viewers see the same volume, it is impossible to present data to one user without other users seeing it. This raises privacy issues, especially if one user is viewing sensitive information. One solution which we implemented was to password encode users' answers in the experiment. We also enhanced the system with audio feedback, since visual feedback could be distracting for users for which the feedback did not apply. Although all users heard the same set of sounds, users knew when the sounds were directed to them since it was synchronized with their keystrokes (for example, hearing the successful noise as soon as they press "enter" to submit their answer). Similar observations and solutions have been seen in the tabletop domain [17, 19].

## FUTURE WORK

In this paper, we have gained an understanding of how text orientation affects readability in both single and multiple user volumetric display applications. We have also developed an effective technique for optimizing text orientation to improve reading times for multiple users. However, there are still a number of directions that can be pursued to extend our work.

Firstly, because our implementation of the search phase of our algorithm used a brute-force technique, applications requiring real-time performance would be forced to limit the granularity of the search, reducing the precision of the results. In our experiment, the angles of the text were pre-computed, but an actual implementation would likely require a more efficient real-time algorithm. It is important to note, though, that even if such an algorithm were employed, our experience suggests that the orientation of text should not be updated every time a user's head position changes, but rather only after large viewpoint changes, in order to limit text movement in the display. As such, the value of a real-time implementation may be limited.

Second, there are some interesting extensions to our orientation optimization which could be implemented. In our implementation, the algorithm optimized the average reading time of all viewers. In some cases, this caused individual reading times to actually increase. An added

constraint could be to ensure no individual times increase. Another idea is to optimize a weighted average of the individual reading times. For example, if text were specifically important for two users, the algorithm could assign extra weight to the reading times for those two users.

Furthermore, the divergence from the original plane could be weighted in the average, with more divergence increasing the average. In our implementation, a set maximum divergence was defined. If the divergence were incorporated into a weighted average, then the divergence could exceed such a maximum value in situations where this would drastically reduce reading times.

Our work compliments the work of Bell et al. [3], who present an algorithm for optimizing the *position* (as opposed to *orientation* in our work) of textual layouts in 3D. An obvious extension would be to combine both algorithms, such that the position and orientation of text were optimized for multiple viewers. This would be especially helpful for volumetric displays, since the results of Experiment 2 suggest that the location of text affects the amount of head movement required to read it clearly.

Another direction is to look at alternative methods for presenting text to multiple viewers. For example, multiple copies of text could be provided when appropriate, so each user has a heads-up view, similar to that in Feiner et al. [13]. Another idea is to curve the text in 3D space. Finally, although uppercase letters worked well, and did not reduce reading speeds when the text was not rotated, new fonts could be developed that allow for both upper and lowercase letters while still reducing character ambiguities and improving reading times when viewed from any orientation.

In summary, we have provided important data on the effects of orientation on text readability for the collaborative use of 3D volumetric displays. Based on this data, we have presented a technique to optimize text orientations, which was shown to significantly reduce reading times. These contributions will be valuable for future designers of collaborative applications for volumetric displays.

#### ACKNOWLEDGEMENTS

We would like to thank the members of the DGP lab for helpful discussions, John Hancock, Anand Agarawala, Jack Wang, and Noah Lockwood for helping with the photographs, and the participants of our studies.

#### REFERENCES

1. Agrawala, M., Beers, A., Fröhlich, B., Hanrahan, P., McDowall, I., Bolas, M. (1997). The two-user Responsive Workbench: support for collaboration through individual views of a shared space. *ACM SIGGRAPH*. p. 327-332.
2. Balakrishnan, R., Fitzmaurice, G., Kurtenbach, G. (2001). User interfaces for volumetric displays. *IEEE Computer*. p. 37-45.
3. Bell, B., Feiner, S., Höllner, T. (2001). View management for virtual and augmented reality. *ACM UIST*. p. 101-110.
4. Benko, H., Ishak, E., Feiner, S. (2004). Collaborative mixed reality visualization of an archaeological excavation. *Int. Symp. on Mixed and Augmented Reality*. p. 132-140.
5. Bowman, D., North, C., Chen, J., Polys, N., Pyla, P., Yilmaz, U. (2003). Information-rich virtual environments: theory, tools, and research agenda. *ACM VRST*. p. 81-90.
6. Brown, L., Hua, H., Gao, C. (2003). A widget framework for augmented interaction in SCAPE. *ACM UIST*. p. 1-10.
7. Buxton, W., Fitzmaurice, G. (1998). HMD's, Caves, & Chameleon: A human-centric analysis of interaction in virtual space. *Computer Graphics*, 32(4). p. 64-68.
8. Chen, J., Pyla, P., Bowman, D. (2004). Testbed evaluation of navigation and text display techniques in an information-rich virtual environment. *IEEE Virtual Reality*. p. 27-31.
9. Cruz-Neira, C., Sandin, J., DeFanti, T., Kenyon, R., Hart, J. (1992). The CAVE: audio visual experience automatic virtual environment. *Comm. of the ACM*, 35(6). p. 64-72.
10. Cruz-Neira, C., Sandin, J., DeFanti, T. (1993). Surround-screen projection-based virtual reality: the design and implementation of the CAVE. *ACM SIGGRAPH*. p. 135-142.
11. Cutler, L., Frölich, B., Hanrahan, P. (1997). Two-handed direct manipulation on the responsive workbench. *ACM I3D*. p. 107-114.
12. Ebert, D., Bedwell, E., Maher, S., Smoliar, L., Downing, E. (1999). Realizing 3D visualization using crossed-beam volumetric displays. *Comm. of the ACM*, 42(8). p. 101-107.
13. Feiner, S., MacIntyre, B., Haupt, M., Solomon, M. (1993). Windows on the world: 2D windows for 3D augmented reality. *ACM UIST*. p. 145-155.
14. Fitzmaurice, G., Balakrishnan, R., Kurtenbach, G. (1999). An exploration into supporting artwork orientation in the user interface. *ACM CHI*. p. 167-174.
15. Grossman, T., Balakrishnan, R. (2006). An evaluation of depth perception on volumetric displays. *AVI 2006*. p. 193-200.
16. Grossman, T., Wigdor, D., Balakrishnan, R. (2004). Multi-finger gestural interaction with 3D volumetric displays. *ACM UIST*. p. 61-70.
17. Hancock, M., Shen, C., Forlines, C., Ryall, K. (2005). Exploring non-speech auditory feedback at an interactive multi-user tabletop. *Graphics Interface*. p. 41-50.
18. Larson, K., van Dantzich, M., Czerwinski, M., Robertson, G. (2000). Text in 3D: some legibility results. *ACM CHI Extended Abstracts*. p. 145-146.
19. Shoemaker, G. B. and Inkpen, K. M. (2001). Single display privacyware: augmenting public displays with private information. *ACM CHI*. p. 522-529.
20. Tinker, M. A., (1965). Bases for effective reading. Minneapolis: Lund Press, Inc.
21. Wigdor, D., Balakrishnan, R. (2005). Empirical investigation into the effect of orientation on text readability in tabletop displays. *ECSCW*. p. 205-224.