2012

Evaluation of gesture based interfaces for medical volume visualization tasks

Cam Kirmizibayrak
George Washington University

Nadezhda Radeva
George Washington University

Mike Wakid
George Washington University

John W. Philbeck
University of Wollongong, johnp@uow.edu.au

John Sibert
George Washington University

See next page for additional authors

Evaluation of gesture based interfaces for medical volume visualization tasks

Abstract

Interactive systems are increasingly used in medical applications with the widespread availability of various imaging modalities. Gesture-based interfaces can be beneficial to interact with these kinds of systems in a variety of settings, as they can be easier to learn and can eliminate several shortcomings of traditional tactile systems, especially for surgical applications. We conducted two user studies that explore different gesture-based interfaces for interaction with volume visualizations. The first experiment focused on rotation tasks, where the performance of the gesture-based interface (using Microsoft Kinect) was compared to using the mouse. The second experiment studied localization of internal structures, comparing slice-based visualizations via gestures and the mouse, in addition to a 3D Magic Lens visualization. The results of the user studies showed that the gesture-based interface outperform the traditional mouse both in time and accuracy in the orientation matching task. The traditional mouse was the superior interface for the second experiment in terms of accuracy. However, the gesture-based Magic Lens interface was found to have the fastest target localization time. We discuss these findings and their further implications in the use of gesture-based interfaces in medical volume visualization, and discuss the possible underlying psychological mechanisms why these methods can outperform traditional interaction methods.

Keywords

visualization, tasks, gesture, volume, evaluation, medical, interfaces

Disciplines

Education | Social and Behavioral Sciences

Authors

Cam Kirmizibayrak, Nadezhda Radeva, Mike Wakid, John W. Philbeck, John Sibert, and James Hahn

Evaluation of Gesture Based Interfaces for Medical Volume Visualization Tasks

Can Kirmizibayrak[1,3], Nadezhda Radeva[1], Mike Wakid[1], John Philbeck[2], John Sibert[1], James Hahn[1]

[1] Department of Computer Science, The George Washington University, Washington, DC, USA
[2] Department of Psychology, The George Washington University, Washington, DC, USA
[3] Department of Radiation Oncology, Stanford School of Medicine, Stanford, CA, USA

Abstract— Interactive systems are increasingly used in medical applications with the widespread availability of various imaging modalities. Gesture-based interfaces can be beneficial to interact with these kinds of systems in a variety of settings, as they can be easier to learn and can eliminate several shortcomings of traditional tactile systems, especially for surgical applications. We conducted two user studies that explore different gesture-based interfaces for interaction with volume visualizations. The first experiment focused on rotation tasks, where the performance of the gesture-based interface (using Microsoft Kinect) was compared to using the mouse. The second experiment studied localization of internal structures, comparing slice-based visualizations via gestures and the mouse, in addition to a 3D Magic Lens visualization. The results of the user studies showed that the gesture-based interface outperform the traditional mouse both in time and accuracy in the orientation matching task. The traditional mouse was the superior interface for the second experiment in terms of accuracy. However, the gesture-based Magic Lens interface was found to have the fastest target localization time. We discuss these findings and their further implications in the use of gesture-based interfaces in medical volume visualization, and discuss the possible underlying psychological mechanisms why these methods can outperform traditional interaction methods.

Index Terms — gesture-based interfaces, human- computer interaction, medical visualization, user study, volume visualization.

I. INTRODUCTION

The quality of health care depends crucially on the ease and success with which physicians are able to construct accurate mental representations of one or more registered 3D imaging

modalities, such as CT and MRI scans, depicted on the 2D displays of traditional computer monitors [1, 2]. The consequences of failing to correctly interpret medical datasets can be devastating, not only in terms of the human costs, but also in terms of the financial burden incurred by the patient and by the health care system. The common practice of visualizing 3D modalities as a series of 2D slices requires physicians to mentally reconstruct the volume by cognitively merging the 2D images, thereby building a 3D mental object [3, 4]. The task of reconstructing and understanding the spatial relationships between external and internal parts of the objects is challenging. Therefore, 3D medical visualizations that allow displaying and interaction of CT and MRI data are widely used in training and diagnosis [5]. However, perceiving the depth relationship among three-dimensional objects on a 2D screen presents a major challenge in the field of medical visualizations. Some of the issues of such visualizations are volume occlusion and ambiguity (or absence) of depth cues.

To help alleviate this problem, various manipulation tools have been created and tested. Examples include changing the viewpoint to use parallax effects or changing rendering parameters such as transparency. Some of the most widely used interaction interfaces are the mouse, electromagnetic and optical trackers. The use of real-world objects, referred to as "props", virtual objects guided by the user [6], and controllable animations [7] have been proven effective in perceiving the anatomy and depth relationship between objects. Even though these interaction methods have been shown to be accurate and helpful in volume visualizations, they present challenges when introduced in surgical systems used in the operating room (OR).

In most medical applications, interactivity is often through 2 degrees of freedom devices, such as the mouse. The high level of user familiarity with the mouse might make mouse-controlled interaction methods more effective than other, newer methods. On the other hand, the mouse is a 2D interaction tool and this requires the user to mentally transform the 2D mouse-space onto the three dimensions of the dataset—a transformation that adds to the cognitive load and thus could negatively impact performance. Also, mouse-controlled systems must be sterilized for use in the operating room and their mere presence may disrupt the surgical workflow. Thus, there is both a need to critically

evaluate the utility of mouse-controlled interactivity in the context of medical datasets, and also a need to develop interaction methods that are not subject to the drawbacks of the mouse. Furthermore, these sterilization and workflow concerns apply to tactile user interfaces in general, and touchless interfaces can prove to be a solution that overcomes these problems.

In this paper we study the prospect of using hand gestures acquired using depth cameras (Microsoft Kinect) in manipulating 3D medical visualizations. We compared accuracy and interaction speed of Kinect with respect to the same parameters measured when using the mouse. We conducted two user studies, which aimed to study the performance of interaction methods in rotation tasks and understanding the internal structures in a volumetric dataset. This paper is an extended and revised version of our previous work [8].

II.   RELATED WORK

2.1 Interaction with Volume Visualizations

In earlier work, Hauptmann [9] suggested that users prefer to use gestures and speech when interacting with computer images. The most used interactions were interaction with several fingers at the same time, moving both hands in all three degrees of freedom for performing various tasks, and ultimately using a combination of fingers, hands and several words to execute any desired task. The preference for multimodal interfaces that combine gesture and speech in order to improve the speed and accuracy of interactions with 3D virtual objects have been investigated and confirmed [10-12].

Two-handed interactions for 3D volume manipulations have been observed to be more natural and better performed when implemented with the user and the task in mind [6, 13, 14]. Zeleznik et al. developed various two-handed cursor techniques for 3D object transformation by separating the tasks done by the dominant hand and the non-dominant hand. The resulting parcellation of degrees of freedom across hands helped in making the tasks of translation, rotation and scaling more natural and precise. Similar two-handed interactions can be used to manipulate the position of the camera in the 3D world. Such interactions contribute to the execution of tasks that are difficult to do using a single mouse or tasks that require great concentration and high precision. However, the authors note that such a system has a few disadvantages: there are physical constraints of the devices depending on whether relative or absolute input devices are used.

Hinckley et al. [6, 14] discuss several useful guidelines when designing an interactive system. For example, they propose that not all degrees of freedom should be used in performing a task if not all of them are necessary. Other aspects of this summary of features of interactive interfaces is the use of clutching mechanisms to help with the execution of the task, ergonomic designs to make the system easier to use without introducing fatigue, introducing transparency or other rendering techniques to help with orientation, and adequate choice of interface tracking method.

Until recently most gesture systems would rely on trackers or some other technology that would require wires running to the computer ports. However, Breuer et al. [15] and Ahn et al. [16] point out two of the most important principles in Human Computer Interaction systems – the action of the user should not be invasive and the interaction device should not disturb the user. This makes many of the methods used in motion tracking inadequate because of the nature of the technology and the wires connecting the device to the computer. Breuer et al. were one of the first researchers to develop non-invasive tracking of hands using a time of flight camera. At the early stage of this study, the system was subject to problems with visibility, occlusion, and data points that could not be tracked in consecutive frames. Ahn et al. also used a time of flight camera tracking system combined with the Kalman filter in order to distinguish the position and depth of the user's hands and implement a computer system for simple manipulations of 3D objects. However, no user studies have been done to validate the importance and the adequacy of the system and its usability.

In the past couple of years more attention has been paid to the integration of game controllers like the Wii remote [17, 18] and the XBox game controller in the manipulation of 3D objects and tracking of the user. Ardito et al. [19] conducted a user study to test the performance of users when manipulating 3D virtual objects using three different methods: the mouse, Wiimote and the XBox controller. Every user had to perform object rotation, translation and path. The controlled study shows that the two game controllers are outperformed by the standard mouse when precision is taken into consideration. However, it was noted that the results are dependent on the familiarity of the user with every device, and the Wiimote has the chance of performing best if the user is comfortable with the controller and experienced in properly manipulating it.

2.2 Spatial and Depth Perception

There is a wide range of literature that suggests the help of dynamic systems for effective volume visualization. The spatial cognition of volumetric objects is highly dependent on the rendering mode of the system. Experiments have been developed to study depth perception of 3D visualizations depending on various ways of direct volume rendering techniques [20]. This study showed an increased accuracy in the dynamically rendered volume and improved precision in depth perception by providing a perspective projection.

The importance of dynamic rendering for user interaction has also been shown by Sando et al. [21]. This study confirmed that users perform better if provided with animations of the displayed object and when given the possibility to interact with the object.

Volume renderings displayed on a 2D screen can cause loss of orientation when no reference points are provided. In cases of medical visualization where only one 3D object is shown, the physician can easily lose perception of how the object is oriented. User studies done by Stull et al. [22] show that providing orientation references in the form of the three axes – x, y and z, can help in matching the target orientation. Compared with an experiment where no reference handles have been provided, the study shows that accuracy in matching an

object to a target orientation can be increased while simultaneously decreasing performance time by providing the main three axes of rotation in the object reference frame.

III. GESTURE BASED INTERFACES FOR MEDICAL VOLUME VISUALIZATION

The current standard interaction interfaces for medical applications can be classified in two groups: image-space and object-space. Image-space interactions are performed by interacting with the produced visualization in the virtual world, and are most of the time in 2D. Object space interactions enable the users to interact with the visualization in the real world, and can perform 3D interactions. Currently, the most widely used image-space interfaces use mouse input, while object space interactions are performed by using trackers. Both of these groups have drawbacks, some of which are explained in detail by Johnson et al. [23] from an interventional radiology context. Asepsis must be maintained; therefore the visualization apparatus must be sterilized. Also, physicians can lose attentional focus at crucial time-points if there is a need to move to a different location to interact with the system.
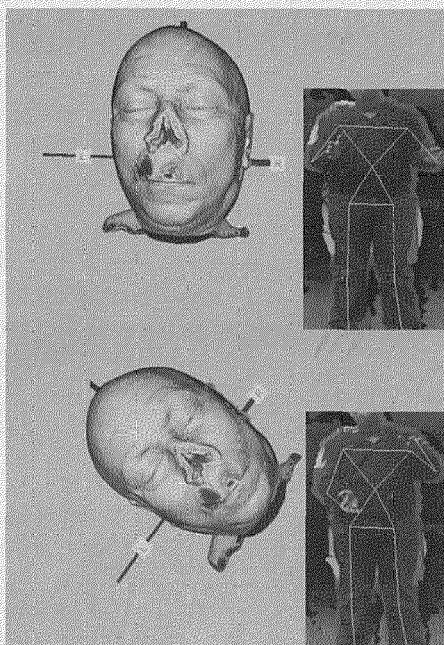

Fig. 1. Rotation by hand locations.

A recently and rapidly developing trend in human-computer interaction is the development of 'natural' user interfaces, which aim to use familiar actions to control systems to improve performance and reduce the time necessary for the users to become acclimated with systems. The use of gestures can provide an alternative that can overcome the aforementioned limitations of tactile interaction interfaces. A few research efforts (for instance Gestix [24]) have used computer vision techniques to track users' hand locations for interaction with

visualizations, but this work focused more on the implementation aspects of designing such systems. The introduction of Microsoft Kinect has provided researchers with a cheap and robust alternative to track user movements and translate them into interactions, with applications in many diverse areas [25]. In this paper, we provide human factors analysis of a Microsoft Kinect based system with the ultimate goal being development of a device useful for use in medical visualization settings, especially in the operating room.

In this section, we will describe some of the interaction methods using Kinect for medical visualization applications. All of these interaction methods used the tracked location of one or both user hands, which were extracted using the OpenNI framework [26]. Our aim in providing these methods here is to describe how they were implemented in our experiments, while the alternatives are presented for comparison purposes.

Rotation

The rotation with two hands can be implemented in a variety of ways. The first method uses the relative locations of two hand positions in the real world. In this method, the vector between the hand positions (right to left in our implementation) was used to calculate the X- and Y- plane rotations of the object. This is shown with examples in Fig. 1. One way to understand this rotation is to imagine that the user is holding the object from its sides (as indicated by L and R in Fig. 1) and performing the rotation accordingly (this was the way the interface was described to the subjects in our experiments).

The three main axes of rotation are used as rotation references. The yellow cubes denoted by L and R in the visualization respectively correspond to left and right hand locations of the user, the correspondence to user's hand locations can be seen in the right side of the figure (dataset credit: OsiriX volume database [27]).

Another way to perform the rotation is by ratcheting the rotation incrementally. For this, we chose the distance between the hands as an engagement/disengagement trigger. While the hands are closer to each other than a threshold (we used the width between the shoulder joints as the threshold) the rotation is performed as described in the previous paragraph. When the hands are spread farther apart, the current rotation is saved and the next rotation is performed by using the current saved rotation as the starting value. However, our experience with this method and the feedback from our pilot study users was that distance based thresholding was neither very intuitive nor robust. We also tried holding hands still as an engagement/disengagement trigger, which sometimes produced undesirable results such as increasing the time to complete tasks.

Scaling

Hand positions can be used to manipulate the size of the object by mapping the zoom factor or object size to the relative distance of the hands.

Gestures as a Mouse

The projected hand position can be used to control the cursor location. For this, we need a trigger for the mouse click: two possibilities are using the dominant or non-dominant hand distance to the camera as triggers. For this, the depth value of hand locations are compared with the depth of another joint (e.g.

right hand with right shoulder), and if the distance is more than a threshold, a mouse button down action is registered.

Changing Viewpoint

We can use hand positions to change the virtual camera location. For this interaction, the vector between two hands is used as the camera look vector (l). To define the camera, we need an up vector(u), for which the up vector of the Kinect camera space (u' = {0 , 1 , 0}) can be used. u and the right vector (r) then can be calculated as in (1)-(2):

$$r = u' \times l \tag{1}$$
$$u = l \times r \tag{2}$$

This method works using the left hand location as the camera origin, while the right hand always looks at the origin. If desired, the camera zoom can be adjusted using the right hand's distance to the left hand.

Slice-based Visualization

2D slices are the most commonly used visualization method for 3D medical datasets because of their familiarity to the radiologists and surgeons and their success in presenting size information since no perspective projection is used. These visualizations can be augmented with 3D renderings where the location of the slice can be shown as a placeholder to help physicians understand the relative location of the current slice to the rest of the dataset. The location of these slices in X, Y and Z planes can be changed by gesture based interactions. In our implementation we used the relative height (y value) of the right hand with respect to the torso location of the user extracted by Kinect to select the slice location on the XZ plane, with the shoulder width used to normalize the height to [0,1] range of the dataset.

3D Magic Lens

The Magic Lens interface was proposed by Bier et al. [28] as spatial filters to "modify the visual appearance of objects, enhance data of interest or suppress distracting information". This was later applied to volume visualization by several researchers [29, 30]. The location of a volumetric Magic Lens can be controlled by a gesture-based interface. The orientation can be either perpendicular to the camera projection plane, or can be changed by moving the non-dominant hand. An example can be seen in Fig. 2(a). In this example, the Magic Lens is used to apply different transfer functions to inside and outside the currently selected region.

Volume Editing

Volume editing operations can be used by moving a volumetric brush to perform accumulated sub-volume selections and changing the rendering parameters inside different selected sub-volumes. The lens location again can be selected by the dominant hand location in 3D. Implementation details for such a rendering approach are beyond the scope of this paper. Interested readers can refer to our previous work [31, 32]. An example volume editing result is shown in Fig. 2(b).
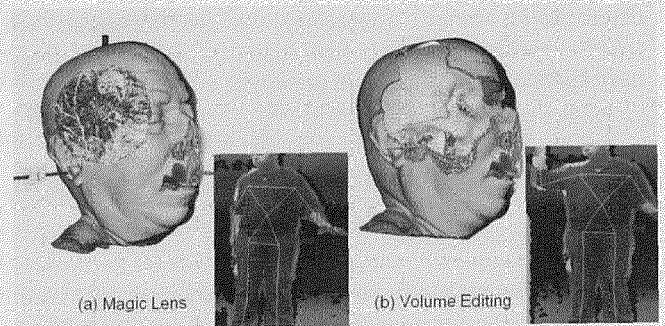

Fig. 2. Examples of Magic Lens visualization and volume editing. The volumetric lens location in (a) and brush location used to create (b) are controlled by the user's right hand, while the editing mode is activated by raising left hand.

The methods presented in this section can be useful for a variety of medical visualization tasks. In order to evaluate the effectiveness of gesture-based interfaces, we have conducted user studies of two of the most widely used tasks for interactive volume visualizations; rotation and finding structures inside a volume. These two volume interactions are a necessary part in the exploration of medical volumes, allowing the physician to study both the overall shape of the object and its inner structures.

IV.   EXPERIMENT I: ROTATION

The first experiment asked the users to rotate a volume rendered visualization to match a target visualization, which was rotated randomly by one of several pre-defined possible orientations. The computer screen was split into two parts. The right part displayed the target orientation of the volume. The left side showed the same volume but in a different orientation. The user had control over the left-side volume and rotated the object so that it matched the orientation shown on the right hand side. In order to limit the amount of time spent by each user, we have limited the possible random rotations to 0, -45 and 45 degrees in the X and Y planes, resulting in 9 possible orientations for each study. These orientations were presented to users in a randomized order. The users performed all 9 matching tasks in one interaction interface before switching to next. The order of the interfaces was also selected randomly. After all possible interfaces were tested, the process was started again for a second test, with an independent random order. Our aim with this randomization of interface orders was to minimize any systematic effects of familiarity and learning by distributing these effects randomly across users, and the analysis of results showed no order effects were present in either of the experiments.
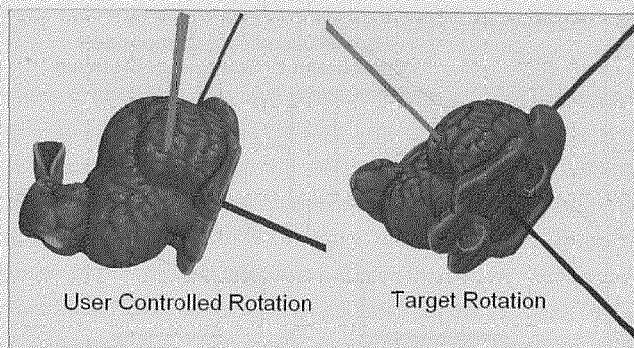
Fig. 3. A sample screen for Experiment I. The users try to match the rotation on the left side of the screen to the target orientation seen on the right.

The subjects were instructed to get close to the target as fast as possible and to indicate when they think they are reasonably close to the target shape. The test administrator then advanced to the next target. We chose not to use an automatic method for advancing to the next trial for two reasons. First, since both hands are used for interaction, the users have no obvious gesture to indicate advancement, and using measures like hand distance to indicate engagement/disengagement was found to be distracting and insufficiently robust in our pre-experiment tests. Second, the use of numerical similarity (e.g. advance when the current difference in X and Y axes are less than 5 degrees for 2 seconds) or the use of changing background color when the orientation is within an acceptable range was found to have fluctuating effects in performance because of the difficulty of perceiving rotations from 3D renderings. In some cases, the user would think s/he was close and wait for auto-advancement when outside the accepted range, or would exhibit overreliance on the background change, which is not a realistic assumption in a real application setting. We believed these advancement methods would introduce undesired errors to our results. Similarly, removing the time limit (instantaneous advancement when the rotations are close) was not used because we believed users could try to randomly change orientations and rely on the software to show them if this was a correct match.

Training

All users provided their informed consent to participate. The experimental protocol was reviewed and approved by the GW ethics committee. A short description of the system was given next. The goals of the experiment were not explained in detail to the users to avoid any bias towards one of the interfaces, except explaining that the performances of different interfaces for some common volume visualization tasks were being compared. In order to avoid data discrepancies due to lack of experience with the system, users were allowed to perform the same interaction tasks as in the real experiment but without their performance recorded. The users trained on a different volumetric dataset (Piggy bank dataset [33]) so that they do not learn the shape (and the internal structure for Experiment II) of the dataset for the experiment, but get familiar with the interaction methods and the nature of the tasks. For the experiment, we used the Stanford Bunny [33], because of its

iconic status and since it provides a known shape with satisfactory variability.

Pilot Study

Before performing our experiment, we conducted a pilot study with five members of our research group (not including the authors). We were especially concerned about the use of gestures to simulate a mouse, since the clicking gesture using a depth threshold felt awkward to use in our pre-experiment tests. Similarly, using the other hand to perform the clicking was thought to be distracting and not very intuitive.

In the pilot study, three interfaces to perform rotation tasks were tested: Kinect two-handed rotation (K2HR), Kinect mouse rotation (KMR) and traditional mouse rotation (TMR). For K2HR, the rotation was performed as explained in Section III, by using the vector from user's right hand to left hand. For TMR, the rotation was performed around the center of gravity of the object (as indicated by the intersection of the displayed axes) using the GLUTManipulator class rotate function [34]. We chose a standard and widely used rotation function to perform mouse rotation to ensure an accurate comparison. The function rotates the object around the center of rotation of the object when the user left-clicks and drags the mouse. The axis of rotation is perpendicular to the direction of the current mouse movement. For KMR, the threshold for clicking was selected to be the plane 40 centimeters closer to the camera than the right shoulder plane (both planes perpendicular to the image plane), while the projected right hand location was used to control the mouse location as described above. This threshold corresponded to a slightly bent elbow for most users, which we believed to be a reasonable distance. The rotation process was otherwise identical to TMR (i.e. the same function was called with the mouse location and clicking actions).

The results of the pilot study reinforced our assumptions that KMR in this implementation would not be comparable to the other two interfaces. The users commented the interface was frustrating to use and the overall time spent on these experiments was significantly higher than on the other two interfaces – average time for KMR was 29.3s versus K2HR time of 14.8s and TMR time of 17.0s . Therefore, we decided to omit KMR from our experiment and decided to proceed with using only TMR and K2HR.

Controlled User Study

The study group consisted of 15 people between the ages of 22 and 38, with an average age of 29.4. Out of the fifteen users, 12 were male and 3 were female. Our subjects were all college-educated adults. None of the subjects indicated they have used a gesture-based interaction interface before, or played with the Microsoft Kinect gaming platform. 7 of the subjects indicated they occasionally use software that produces 3D renderings, while 8 said they never use such software. On average it took about 6 minutes to perform the training and 12 minutes to perform and collect all data. The subjects stood about 6 feet from the monitor (a 42-inch LCD screen) for the Kinect interactions, and a mouse was provided around the same distance to be used standing up to ensure consistency between the interfaces. The monitor orientation was constant across trials, and was set to face the approximate centerline between

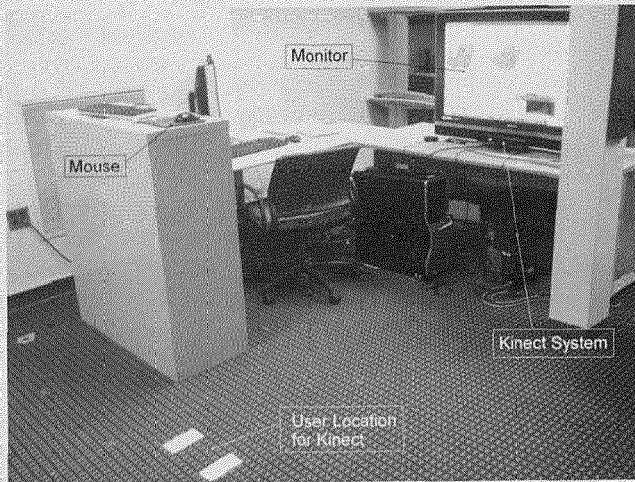mouse and Kinect locations. Our setup can be seen in Fig. 4.

Fig. 4. Experimental Setup.

The authors' expectations in this part of the study were twofold. First, we hypothesized that users would find it easier or of equal difficulty to rotate and match the target orientation when using K2HR than when using TMR. We believed that K2HR was more flexible in trying to quickly re-orient the volume and adjust to the correct orientation. K2HR was also believed to be a more natural interface to use than TMR because it mimics holding an object in one's hands and rotating it in the X- and Y-axes. Second, we expected that the time it takes for the correct matching between the object and target orientation to be comparable between the two orientation interaction methods. However, the TMR interface could prove to be more accurate than K2HR since it is possible that precise small movements could be performed more easily with the mouse.

Data Analysis and Results

The data were analyzed using 2-tailed, paired-sample t-tests. Each subject's mean value across all trials was used for analysis. Accuracy and time were analyzed separately, with the independent variable in each test being the interface type (K2HR or TMR). For the accuracy measure, we used the quaternion representation, with rotation error calculated as the quaternion norm of the difference between target and the user selected rotations. The box plots of the results are displayed in Fig. 5 for rotation error, and in Fig.6 for time.

Accuracy. The rotation error analysis showed that the accuracy in orientation matching with K2HR (average error of 0.09 quaternion units) was significantly higher than the accuracy using TMR (average error of 0.13 units; $t[14] = 3.19$; $p < 0.007$).
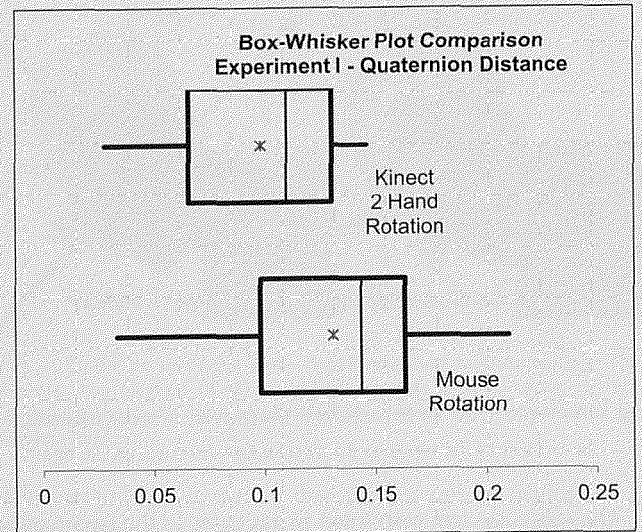

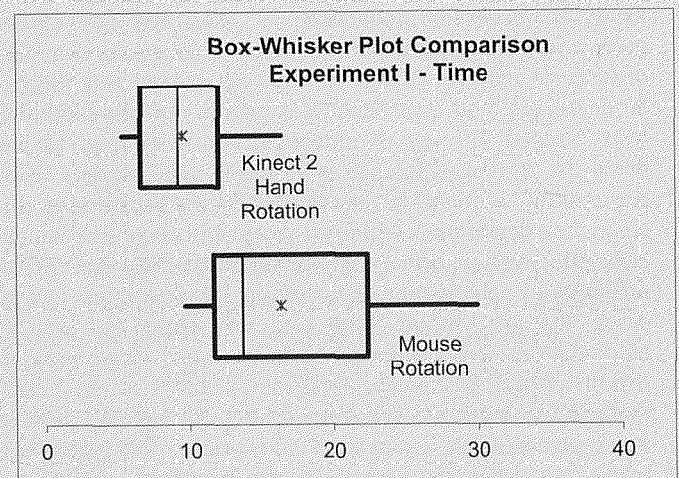Fig. 5. Box plot of results for Experiment I (rotation error).

Fig. 6. Box plot of results for Experiment I (time).

Time. The average time it took users to rotate the object to its target orientation was reliably greater with TMR than with K2HR (16.3s vs. 9.4s, respectively; $t[14] = 5.89$; $p < 0.001$). Furthermore, the between-subject variance in time for K2HR was much lower than for TMR (11.6 vs. 42.8, respectively). This shows that users quickly adjusted to the K2HR interface and tended to perform more similarly to each other using that interface.

In sum, we can conclude that for both testing conditions, performance using the gesture-based interface was reliably better than when using the mouse as a rotation interface. The results are significant in showing that the K2HR is an accurate and fast interface for rotation matching tasks.

V.   EXPERIMENT II: TARGET LOCALIZATION

For this experiment, the task was localizing artificially created targets inside a volume. This task aimed to replicate the way physicians display volumetric datasets in a variety of medical diagnosis and treatment applications – i.e., by studying consecutive 2D slices of the volume to detect abnormalities or changes in the internal structures. Three interfaces were tested: Mouse slice (MS), Kinect slice (KS) and 3D Magic Lens (ML). For MS and KS, the users controlled the location of a cross-section, shown in the right half of the window. The slice location with respect to the volume was also displayed on the left half of the window, along with an opaque volume rendering of the dataset, which did not show any of the target structures within. The ML interface was provided as an alternative interface to explore the inner structures in 3D instead of using 2D slices. The Magic Lens filter was defined as showing the target structures but not the dataset boundary surface, in effect acting as a transparent volumetric sub-region that reveals the inner structures. A cylinder was selected as the lens shape. The boundaries of the lens were indicated by drawing lines showing the shape of the cylinder. The center of the lens (which the subjects are instructed to match to the target) was displayed as a small square on the top of the cylinder.

The targets were created by copying a smaller volume texture to specific locations of the volume dataset, which were defined manually beforehand and were common across all subjects. 10 possible target locations were defined. In each trial, 9 of these were used as distractors, while the remaining one was the target for that trial. The difference between targets and distractors was the size: targets had a radius of 15 voxels while distractors had a radius of 10 voxels. The users were instructed to search the volume by either changing the slice location (using the mouse or Kinect) or moving the Magic Lens around. When the target was located, users were instructed to try to center it and instruct the test administrator to advance to the next trial. Two example results are shown in Fig. 7 and Fig. 8 for the slice-based and Magic Lens interfaces respectively. We used the same kind of advancement method to maintain continuity with our previous experiment.
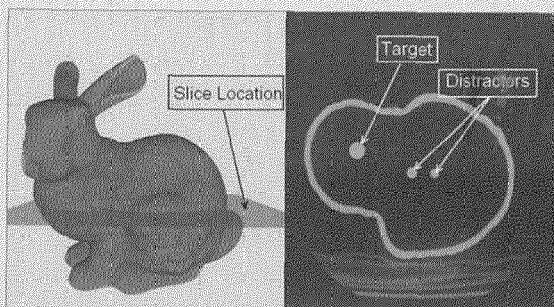

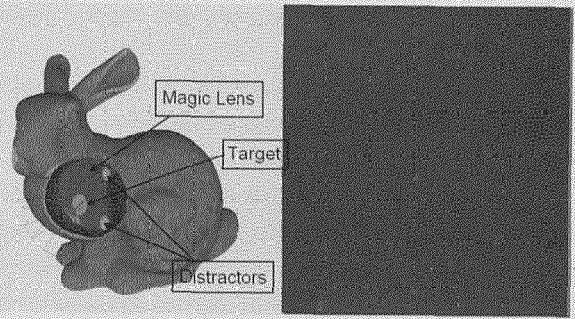Fig. 7. A sample screen for Experiment II (for slice-based visualizations)

Fig. 8. A sample screen for Experiment II (for Magic Lens visualization). Note that the right side of the screen is kept blank intentionally for consistenct with other interfaces, as a 2D slice is not present for the ML interface.

Pilot Study

We again performed a pilot study beforehand with the same group of pilot users. For this experiment, we were concerned if the performance of ML would be comparable to MS and KS, because it would be an unfamiliar interaction method for most of the users, while just changing the slice location using a slider was thought to be a cognitively simpler and more familiar task. Moreover, while ML presented the targets with 3D renderings (and possible size ambiguities because of the perspective projection), MS and KS essentially required the user to infer the 3D shape by looking at 2D slices. However, analyzing the data obtained from the pilot study and the feedback given by the participants, we concluded that ML could be comparable in performance to the other interaction methods. Therefore, we decided to include all three interaction methods in our experiment.

Controlled User Study

This experiment was done with the same group of people who participated in Experiment I. Experiment I and Experiment II were performed sequentially (as were the pilot studies, as we believed the tasks to be reasonably different from each other and learning effects should not affect the performance). The users were allowed to do a training session similar to the one in Experiment I. On average, it took about 3 minutes to perform the training and 6 minutes to perform and collect all data. For training, the target and distracter objects were positioned inside the Piggy Bank dataset and the user was asked to select the largest object using the three interaction modalities. After the training, the user performed the real experiment using the Stanford Bunny with data recording of time and accuracy. Each interface was tested with 5 target locations, and the possible target locations in the training and experiment were different

from each other (one half of the 10 used in training, while the remaining half used for the experiment). After all three interfaces were tested, the experiment was repeated again in an independently random order, in the same manner of Experiment I.

Our hypothesis for the experiment was that users would locate the target object slightly faster when using the MS rather than using KS. The first reason for this expectation is the greater familiarity of the mouse than the KS interface. Secondly, in the MS interaction the user is presented with an on-screen slider, and the movement of the mouse corresponds to the relative position of the slider with respect to its full range. In KS, even though the origin location depends on the location of the torso joint, that information may not be apparent to the user: as far as the users are concerned, they are moving their hands in air without a real frame of reference. Moreover, precise movements could be more difficult with the KS and ML because of the accuracy of the skeleton extraction from depth images. However, based on previous research and our pilot study, we expected to see a close correspondence in performance when using MS or KS and that user's accuracy could be comparable. This would be a good indication of a successful interface that would allow physicians to scan through a series of 2D slices of the patient while in the OR and without any danger of contamination of the physician's hands from the mouse.

We also expected that the time it takes for the user to locate abnormal structures in the inside of the volume when looking at 2D slices would be close to the time it takes to use the Magic Lens interface, since the user is presented with the 3D shape using ML. We aimed to determine if the ML helps the user better understand the internal structures of the volume and the spatial relationship between objects, which could improve the success of pre-operative planning and diagnosis, and volume dataset understanding during surgery.

Data Analysis and Results

We performed two separate 1x3 repeated measures analyses of variance (ANOVA) on the accuracy and time data, with interface type as the repeated measures factor. For accuracy, either the slice distance from the target center (for MS and KS), or the projected pixel distance in Y-axis (for consistency) of the Magic Lens center was used for analysis (for ML). The dataset contained 361 slices, and the volume rendering occupied about 360 pixels on the screen; therefore, a rough equivalence of 1 pixel equaling 1 slice was assumed. Thus, we will use 'units' when discussing error in this section. However, comparing an essentially 2D visualization of slices with the Magic Lens (3D visualization) might complicate this analysis. This possibility will be discussed in more detail in Section VII. The results of

Experiment II are presented as box plots in Fig. 9 and Fig. 10.

The mean performance for KS (7.2s) was on average faster than MS (8.0s). However, the between-subject variance for MS was lower than for KS (3.8 vs. 9.8, respectively). Furthermore, both the error in identifying the target slice and the response variance was lowest for MS. Thus, in general, performance in using MS to select the correct slice was highest along several criteria.

ML is a substantially different interface for studying the internal structure of objects than MS and KS. While ML had a better mean time (6.6s) than either KS or MS, the resulting between-subject variance of 8.0 fell between that of MS and KS (see above for comparisons). Furthermore, the mean error for ML (9.9 units) was higher than for either MS or KS (3.3 vs. 5.3, respectively). The between-subject variance was likewise higher for ML (20, versus 5.7 and 1.0 for KS and MS, respectively). This is not surprising, as we expected the mouse afforded more precise control movements. The high variance in error shows that users did not perform as consistently as a group when interacting with this interface as compared to using the mouse.
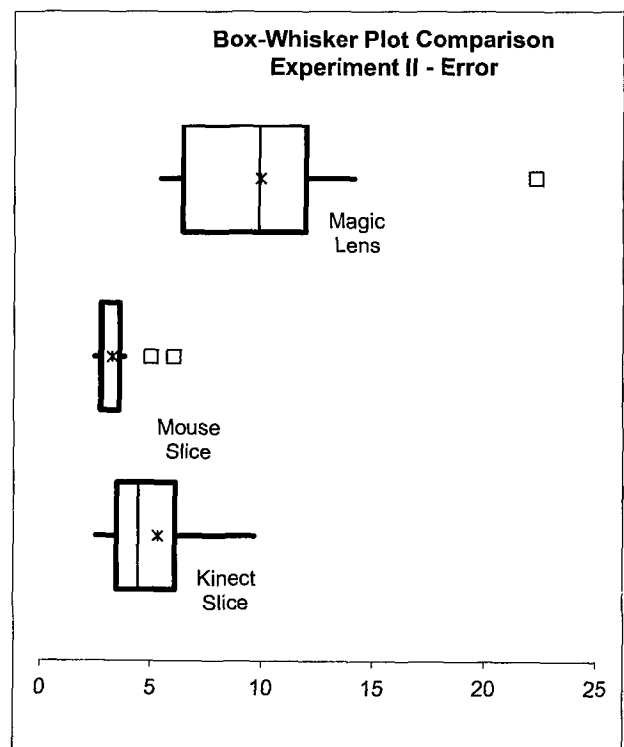

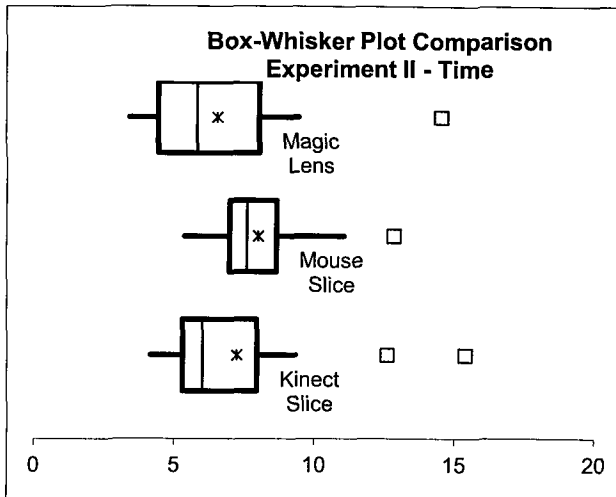Fig. 9. Box plot of results for Experiment II (error).

Fig. 10. Box plot of results for Experiment II (time).

There was a marginal main effect of interface type in the time ANOVA ($F[2, 28] = 3.18$; $p = 0.056$). Pairwise planned comparisons showed that this marginal effect was due solely to differences between the ML and MS conditions ($p = 0.017$; all other comparisons > 0.18). In the error ANOVA, there was a significant main effect of interface type ($F[2, 28] = 19.86$; $p < .0001$). Pairwise planned contrasts showed that performance using either KS or MS was reliably better than performance using ML (both p's < 0.001); performance was only marginally better using MS than for KS ($p = 0.068$). These results suggest that even though ML can be used as an interface for quick exploration of datasets, the mouse is more useful in tasks requiring precise targeting. We discuss possible reasons and further implications of these results in Section VII.

VI. QUALITATIVE RESULTS

The informal feedback from the users was very positive, with many users spontaneously expressing that the Kinect interfaces were interesting and fun to use without being asked by experiment administrators. The users were also asked to fill out a survey to evaluate their experience, the results of which will be discussed in this section.

For the rotation experiment, the K2HR was mostly preferred by the users, with 11 out of 15 (69%) indicating they thought K2HR was easier to use than TMR. When asked which interface helped them understand the shape of the object, an even larger preference (14 out of 15, 93%) towards K2HR was indicated.

The Magic Lens interface was also received favorably by the users. When asked about the ease of use of the interface on a scale of 1 (very easy) to 5 (difficult), mostly very easy (5) and

somewhat easy (6) responses were given, with an average of 2.06 difficulty. Similarly, users responded with an average of 2.13 to the question asking about the ease of exploring the internal structures of the object. Moreover, the users showed a preference toward the ML interface compared to the slice-based visualizations, with 11 out 15 indicating the ML interface helped them understand the internal structure of the object better than KS and MS interfaces. The details of these results are given in Table I.

Comparison of the KS and MS interfaces produced results that are more balanced, with 8 users indicating the KS interface was easier to use compared to 7 for MS. However, 10 out of 15 users said KS slice traversal helped them understand the internal structures of the object better (the synthetic targets in this case).

TABLE 1. MAGIC LENS SURVEY RESULTS.

How easy was it to use the Magic Lens interface:				
Very easy	*Somewhat easy*	*Neutral*	*Somewhat difficult*	*Difficult*
5	6	2	2	0

How easy was it to explore the internal structures of a 3D dataset using the Magic Lens:				
Very easy	*Somewhat easy*	*Neutral*	*Somewhat difficult*	*Difficult*
4	6	4	1	0

Do you think this tool would improve your understanding of 3D datasets and their relation to the real world (e.g. the patient)?		
Yes	Maybe	No
12	3	0

VII. ANALYSIS OF RESULTS

Our experiments have yielded several interesting results. In Experiment I, performance using the gesture-based interface exceeded our expectations. Users performed significantly better using Kinect as compared to the mouse in the rotation task. We believe the success of the interface comes from its similarity to an action that users can relate to (holding and rotating an object), as opposed to the mouse rotation, which is a more abstract mapping. Furthermore, these results were achieved after a short training time using an unfamiliar interface, which points to the intuitiveness of using gestures for rotation tasks. However, because the interface presented here can only handle rotations in two degrees of freedom, it would be interesting to see if users can perform rotations with three degrees of freedom with other gesture-based interfaces (such as using ratcheting as mentioned in Section III). We believe gesture recognition methods that can detect hand gestures such as grasping would be very beneficial for such tasks and can

extend the possibilities of tasks that can be handled with these kinds of interfaces.

In the second experiment, the mouse outperformed both gesture-based interfaces in terms of accuracy, which was an expected result given the suitability of the mouse in making precise movements. For the Magic Lens interface, some other factors might have contributed to the high error rate. In slice-based visualizations, an accurate match requires the slice to be in an exact position since only a cross-section of the data is displayed. However, even though Magic Lens is not perfectly centered at the target location, the target might be inside the lens volume and completely visible. Furthermore, due to perspective projection, the orientation of the Magic Lens might change depending on its location, making it more difficult to center it exactly on the target. These factors, combined with the fact that gesture-based interfaces outperformed the mouse in terms of time makes us believe that gesture based interfaces might be particularly beneficial for exploration of volumetric datasets. Another interesting follow-up to this study might be studying the effects of the precision of the Kinect interface: for instance making larger gestures necessary for the same amount of virtual movement of objects, allowing the non-dominant hand to perform coarse movements while the dominant hand is used for refined movements as presented by Bier et al. [28], or applying smoothing filters, to see the effect of these to the time and accuracy of tasks. Moreover, the Magic Lens interface was received favorably by users, and the fact that it can present the inner structures of the dataset in 3D means that it can contribute to understanding of medical datasets and shapes of internal structures. Furthermore, users could locate targets more quickly with Magic Lens, thus in situations where the user has to compare information between several spatial locations (e.g. if the experiment had more than one target with varying sizes larger than the distractors), the Magic Lens can prove to be effective for quick spatial exploration.

Another interesting result of Experiment I was the fact that the subjects were more accurate as well as faster, even though as Experiment II suggests the mouse interface might be better for precise movements. Several factors might have contributed to this result. The first is more technical: in Experiment II most of the interactions were made with the hand around the area between the shoulder and camera (making the hand almost perpendicular to the camera), which might sometimes cause problems in the accuracy of the pose extraction algorithm used in Kinect. To alleviate this, working space location and arm poses used should be considered in designing gesture based interaction systems. A second possibility comes from the fact that users indicated when to advance in the trials. This result might be interpreted as that the users actually understood that they had a more accurate understanding of how close they were

to the target orientation using the gesture-based interface. This might possibly be helped by using the cues presented by their inherent knowledge of relative locations of their hands, while with the mouse interface they only have a visual feedback to decide whether they are close to the target orientation. Yet another possible factor is that the mouse interface was simply more difficult to use for matching rotations, and the users were more likely to be frustrated and advance to the next trial even though a good match was not achieved. All of these factors could be interesting to study in future research and interface design.

VIII.   DISCUSSION AND FUTURE WORK

Even though our experiments yielded promising results, there are many questions to be answered for optimal use of gesture-based interfaces in particular, and interactive systems in general in medical applications. The prevailing zeitgeist holds that interactivity should improve interpretation of medical datasets. However, research shows that interactivity does not universally help. In some contexts, users who lack domain knowledge or who have low spatial ability do not benefit from interactivity [35, 36]. Much remains unknown about the cognitive factors that link expertise, spatial ability and interactivity. Similarly for gesture-based methods, future work about what kind of gestures are most useful for operating room (OR) physicians for minimizing their cognitive workload is crucial. These gestures must be feasible within the constraints of standard OR practices and also functional within the limits of motion-tracking and gesture analysis technology.

Gesture-based interaction offers several potential benefits for interactivity over and above the mouse. Physical movement can improve mental rotation performance [37], so on the plausible assumption that our tasks draw upon some of the same cognitive processes as mental rotation, gesture-based interactions should facilitate the ability to construct 3D mental models on the basis of medical datasets. In particular, relative to using a mouse, gesture-based interactivity generally involves larger-scale movements and can accommodate using 2 hands rather than being restricted to 1 hand. Both of these factors could engage the processing advantages of physical motion more than mouse-based interactivity. Therefore, the study about the effects of varying movement scales and spatial ability would yield clear findings regarding the possible benefit of increasing the amount of body motion (by using both hands and/or larger-scale movements) for constructing 3D mental models on the basis of medical datasets. If either or both of these factors enhance performance, this would provide clear confirmation of the commonly-held intuition that the body

motions underlying gesture-based interactions have a tangible benefit. If neither factor impacts performance, this would lead to the surprising conclusion that the magnitude of body motion when interacting with a simulated medical dataset plays little role. Of particular interest is whether low-spatial individuals benefit more from increased body motion than do high-spatial individuals. Because people with low spatial ability are thought to have difficulty with mental transformations [38], they potentially have the most to gain from gesture-based interaction. If this idea is confirmed, it would suggest a likely mechanism for the observed benefits of gesture-based interaction: when mental transformations are difficult, the transformations can be externalized through physical body motion, thereby freeing up cognitive resources to improve performance. In the unlikely event that there are no differences dependent on spatial ability, this would provide valuable evidence that constructing 3D mental models on the basis of medical datasets does not draw upon the cognitive processes underlying mental rotation ability.

IX.   CONCLUSION

In this paper, we presented and evaluated effectiveness of gesture-based interfaces to be used in volume dataset rotation and exploration tasks. The gesture-based interface outperformed the mouse for rotation tasks. For locating targets, the mouse outperformed gesture-based interfaces in terms of accuracy but not in terms of time. We hope these results will provide insight for further exploration of these kinds of interfaces, as their use in the operating room might solve several problems currently encountered. Especially, methods to increase the precision of gesture-based interfaces and smooth ways to perform engagement/disengagement actions would be valuable. When designing these kinds of gestures, the limited space of the operating room, accurate differentiations between gestures to interact with the system and expressive or explanatory gestures, and maintaining the surgical workflow need to be considered.

The experiments presented in this paper were designed to study the effectiveness of gesture-based interfaces with medical applications in mind. However, we wanted to have abstract datasets and non-medically trained subjects in this stage of our research to establish the intuitiveness of these kinds of interfaces to users unfamiliar with them (as we expect most physicians will be). We are aiming to conduct similar studies using abstract datasets, as well as using medical datasets and medically relevant tasks with medically trained professionals of different levels of expertise to see if these results translate to real medical applications. We believe studies of effects of inherent abilities on performance using gesture-based

interfaces are also a valuable avenue of research.

REFERENCES

[1]   J. Klein, O. Friman, M. Hadwiger, B. Preim, F. Ritter, A. Vilanova *et al.*, "Visual computing for medical diagnosis and treatment," *Computers & Graphics*, 33(4), pp. 554-565, 2009.

[2]   D. P. Perrin, N. V. Vasilyev, P. Novotny, J. Stoll, R. D. Howe, P. E. Dupont et al., "Image Guided Surgical Interventions," *Current problems in surgery*, 46(9), pp. 730-766, 2009.

[3]   H. Fuchs, M. Levoy, and S. M. Pizer, "Interactive Visualization of 3d Medical Data," Computer, 22(8), pp. 46-51, Aug, 1989.

[4]   S. Dev, "Imaging and visualization in medical education," *IEEE Computer Graphics and Applications*, 19(3), pp. 21-+, May-Jun, 1999.

[5]   C. Imielinska, and P. Molholt, "Incorporating 3D virtual anatomy into the medical curriculum," *Communications of the ACM*, 48(2), pp. 49-54, Feb, 2005.

[6]   K. Hinckley, R. Pausch, J. C. Goble, and N. F. Kassell, "Passive real-world interface props for neurosurgical visualization," in Proceedings of *the SIGCHI conference on Human factors in computing systems: celebrating interdependence*, Boston, Massachusetts, United States, 1994, pp. 452-458.

[7]   M. Betrancourt, "The Animation and Interactivity Principles in Multimedia Learning," *The Cambridge handbook of multimedia learning*, Mayer, ed., pp. 287-296, New York, 2005.

[8]   C. Kirmizibayrak, N. Radeva, M. Wakid, J. Philbeck, J. Sibert, and J. Hahn, "Evaluation of gesture based interfaces for medical volume visualization tasks," in Proceedings of *the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry*, Hong Kong, China, 2011, pp. 69-74.

[9]   A. G. Hauptmann, "Speech and gestures for graphic image manipulation," *SIGCHI Bull.*, 20(SI), pp. 241-245, 1989.

[10] R. A. Bolt, "\“Put-that-there\”: Voice and gesture at the graphics interface," *SIGGRAPH Comput. Graph.*, 14(3), pp. 262-270, 1980.

[11] S. Oviatt, "Multimodal interfaces for dynamic interactive maps," in Proceedings of *the SIGCHI conference on Human factors in computing systems: common ground*, Vancouver, British Columbia, Canada, 1996, pp. 95-102.

[12] M. E. Latoschik, "A gesture processing framework for multimodal interaction in virtual reality," in Proceedings of *the 1st international conference on Computer graphics, virtual reality and visualisation*, Camps Bay, Cape Town, South Africa, 2001, pp. 95-100.

[13] R. C. Zeleznik, A. S. Forsberg, and P. S. Strauss, "Two pointer input for 3D interaction," in Proceedings of *the 1997 symposium on Interactive 3D graphics*, Providence, Rhode Island, United States, 1997, pp. 115-120.

[14] K. Hinckley, J. Tullio, R. Pausch, D. Proffitt, and N. Kassell, "Usability analysis of 3D rotation techniques," in Proceedings of *the 10th annual ACM symposium on User interface software and technology*, Banff, Alberta, Canada, 1997, pp. 1-10.

[15] P. Breuer, C. Eckes, and S. Müller, "Hand Gesture Recognition with a Novel IR Time-of-Flight Range Camera–A Pilot Study," *Computer Vision/Computer Graphics Collaboration Techniques*, Lecture Notes in Computer Science A. Gagalowicz and W. Philips, eds., pp. 247-260: Springer Berlin / Heidelberg, 2007.

[16] Y.-K. Ahn, Y.-C. Park, K.-S. Choi, W.-C. Park, H.-M. Seo, and K.-M. Jung, "3D spatial touch system based on time-of-flight camera," *WSEAS Trans.* Info. Sci. and App., 6(9), pp. 1433-1442, 2009.

[17] L. Gallo, G. D. Pietro, and I. Marra, "3D interaction with volumetric medical data: experiencing the Wiimote," in Proceedings of *the 1st international conference on Ambient media and systems*, Quebec, Canada, 2008, pp. 1-6.

[18] J. C. Lee, "Hacking the Nintendo Wii remote," *IEEE Pervasive Computing*, 7(3), pp. 39-45, Jul-Sep, 2008.

[19] C. Ardito, P. Buono, M. F. Costabile, R. Lanzilotti, and A. L. Simeone, "Comparing low cost input devices for interacting with 3D Virtual Environments." pp. 292-297.

[20] C. Boucheny, G. P. Bonneau, J. Droulez, G. Thibault, and S. Ploix, "A Perceptive Evaluation of Volume Rendering Techniques," *Acm Transactions on Applied Perception*, 5(4), pp. -, Jan, 2009.

[21] T. Sando, M. Tory, and P. Irani, "Effects of animation, user-controlled interactions, and multiple static views in understanding 3D structures," in Proceedings of *the 6th Symposium on Applied Perception in Graphics and Visualization*, Chania, Crete, Greece, 2009, pp. 69-76.

[22] A. T. Stull, M. Hegarty, and R. E. Mayer, "Getting a handle on learning anatomy with interactive three-dimensional graphics," *Journal of Educational Psychology*, 101(4), pp. 803-816, 2009.

[23] R. Johnson, K. O'Hara, A. Sellen, C. Cousins, and A. Criminisi, "Exploring the potential for touchless interaction in image-guided interventional radiology," in Proceedings of *the 2011 annual conference on Human factors in computing systems*, Vancouver, BC, Canada, 2011, pp. 3323-3332.

[24] J. Wachs, H. Stern, Y. Edan, M. Gillam, C. Feied, M. Smith et al., "Gestix: A Doctor-Computer Sterile Gesture Interface for Dynamic Environments," *Soft Computing in Industrial Applications*, Advances in Soft Computing A. Saad, K. Dahal, M. Sarfraz et al., eds., pp. 30-39: Springer Berlin / Heidelberg, 2007.

[25] J. Tanz, "Kinect Hackers Are Changing the Future of Robotics," *Wired Magazine*, no. 06/20/2011, 2011.

[26] O. Organization. "OpenNI User Guide," 05/01, 2011; http://www.openni.org/documentation.

[27] F. OsiriX. "DICOM sample image sets," 06/01, 2011; http://pubimage.hcuge.ch:8080/.

[28] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose, "Toolglass and magic lenses: the see-through interface," in Proceedings of *the 20th annual conference on Computer graphics and interactive techniques*, Anaheim, CA, 1993, pp. 73-80.

[29] J. Viega, M. J. Conway, G. Williams, and R. Pausch, "3D magic lenses," in Proceedings of *the 9th annual ACM symposium on User interface software and technology*, Seattle, Washington, United States, 1996.

[30] C. Kirmizibayrak, M. Wakid, S. Bielamowicz, and J. Hahn, "Interactive Visualization for Image Guided Medialization Laryngoplasty," in *Computer Graphics International 2010*, Singapore, 2010.

[31] C. Kirmizibayrak, M. Wakid, Y. Yim, and J. Hahn, "Interactive Focus+Context Medical Data Exploration and Editing," *Computer Animation and Virtual Worlds*, In Submission, 2011.

[32] C. Kirmizibayrak, "Interactive Volume Visualization and Editing Methods for Surgical Applications," Ph.D. Dissertation, Department of Computer Science, The George Washington University, Washington, DC, 2011.

[33] S. Roettger. "The Volume Library," August, 2011; http://www9.informatik.uni-erlangen.de/External/vollib/.

[34] C. Everitt. "Adaptors to GLUT native commands for manipulator objects," August, 2011; http://code.google.com/p/twars/source/browse/trunk/twars/common/nvGlutManipulators.hh?spec=svn2&r=2.

[35] M. Keehner, M. Hegarty, C. Cohen, P. Khooshabeh, and D. R. Montello, "Spatial Reasoning With External Visualizations: What Matters Is What You See, Not Whether You Interact," *Cognitive Science*,32(7), pp. 1099-1132, 2008.

[36] P. Khooshabeh, and M. Hegarty, "Inferring Cross-Sections: When Internal Visualizations Are More Important Than Properties of External Visualizations," *Human-Computer Interaction*, 25(2), pp. 119 - 147, 2010.

[37] A. Wohlschläger, and A. Wohlschläger, "Mental and Manual Rotation," *Journal of Experimental Psychology: Human Perception and Performance*, 24(2), pp. 397-412, 1998.

[38] C. E. Bethellfox, and R. N. Shepard, "Mental Rotation - Effects of Stimulus Complexity and Familiarity," *Journal of Experimental Psychology-Human Perception and Performance*, 14(1), pp. 12-23, Feb, 1988.

Can Kirmizibayrak, Ph.D., is currently a postdoctoral research fellow at Stanford University. He received his bachelor's degree in Electrical-Electronics Engineering from Bogazici University, Turkey in 2003, his master's degree in Telecommunications and Computers in 2005, and his Ph.D. in Computer Science in 2011; both from The George Washington University. His research interests are medical visualization, image-guided surgery and radiation therapy, interaction in biomedicine and computer graphics.

Nadezhda Radeva is a Ph.D. student in Computer Science at the George Washington University. She received her Bachelor's degree in Computer Science and Astrophysics from Connecticut College in 2009, and her Master's degree in Computer Science from the George Washington University in 2011. Her research focuses on medical visualizations and depth perception. Her research interests also include medical simulations, scientific 3D modeling and interaction.

Mike Wakid, Ph.D., received his bachelor's degree in Computer Science from the University of Maryland, College Park in 2003, his M.S. in Computer Science from The George Washington University in 2005, and his Ph.D. in Computer Science from The George Washington University in 2011. His research interests include medical visualization, image-guided surgery, real-time rendering, and animation.

John Philbeck, Ph.D., is an Associate Professor of Psychology at the George Washington University. He obtained a PhD in Psychology at the University of California, Santa Barbara, in 1997. After a 3-year postdoctoral position at Carnegie Mellon University, he joined the faculty at GW in 2000. He became Director of the Cognitive Neuroscience Program in 2009 and is the Charter Director of the GW Mind-Brain Institute. He is also Research Associate Professor in the GW Department of Neurological Surgery. His research focuses on human visual space perception and navigation.

John Sibert, Ph.D., is a Professor Emeritus of Computer Science at the George Washington University. He received his A.B. in History, Wittenberg University, Springfield, Ohio in 1968, his M.A. in Geography, Miami University, Oxford, Ohio in 1971 and his Ph.D. in

Geography, The University of Michigan, Ann Arbor, Michigan in 1974. He was the founding program chair of ACM User Interface Software Symposium and program chair of ACM/SIGGRAPH Emerging technologies in August 2007. His research interests are Human Computer Interaction, computer graphics, object oriented paradigms and programming languages.

James Hahn, Ph.D., is currently a full Professor in the Department of Computer Science at the George Washington University where he has been a faculty since 1989. He is the founding director of the Institute for Biomedical Engineering and the Institute for Computer Graphics. His areas of interests are: medical simulation, image-guided surgery, medical informatics, visualization, and motion control. He received his Ph.D. in Computer and Information Science from the Ohio State University in 1989, an M.S. in Physics from the University of California, Los Angeles in 1981, and a B.S. in Physics and Mathematics from the University of South Carolina in 1979.