

Visualization of Off-Surface 3D Viewpoint Locations in Spatial Augmented Reality

Matt Adcock^{*†‡}

matt.adcock@csiro.au

^{*}CSIRO
Canberra, Australia

David Feng^{*†}

david.feng@csiro.au

[†]Australian National University
Canberra, Australia

Bruce Thomas[‡]

bruce.thomas@unisa.edu.au

[‡]University of South Australia
Mawson Lakes, Australia

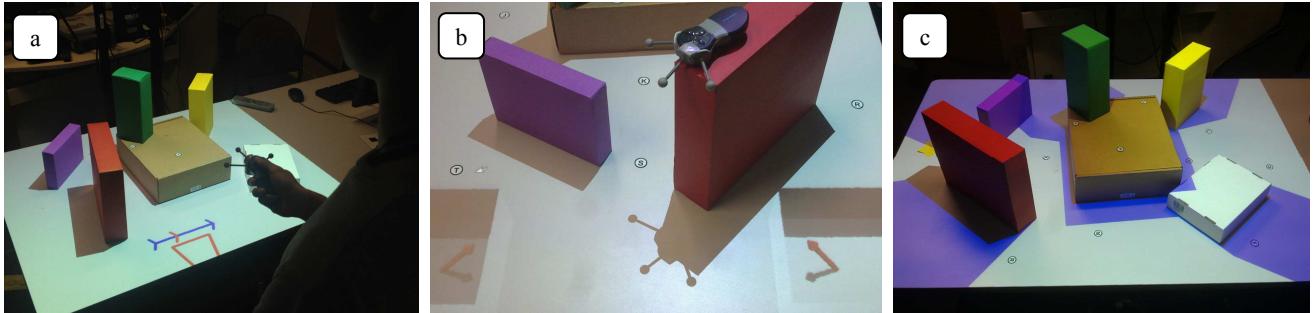


Figure 1. An overview of the techniques we designed to visualize off-surface information. In (a) the red Wedge tracks the location of a user's hand within the display plane and a blue arrow indicates relative height above the surface, in (b) two vectors appear to triangulate the location of the tracked device, and in (c) blue shadows are cast from an Eyelight above the centre of the table.

ABSTRACT

Spatial Augmented Reality (SAR) systems can be used to convey guidance in a physical task from a remote expert. Sometimes that remote expert is provided with a single camera view of the workspace but, if they are given a live captured 3D model and can freely control their point of view, the local worker needs to know what the remote expert can see. We present three new SAR techniques, *Composite Wedge*, *Vector Boxes*, and *Eyelight*, for visualizing off-surface 3D viewpoints and supporting the required workspace awareness. Our study showed that the Composite Wedge cue was best for providing location awareness, and the Eyelight cue was best for providing visibility map awareness.

Categories and Subject Descriptors

H.5.1 [Information interfaces and presentation]: Artificial, augmented, and virtual realities; H.5.2 [Information interfaces and presentation]: User Interfaces – Graphical user interfaces.

Keywords

Spatial Augmented Reality; Spatial User Interfaces; Remote Guidance; Visualization; Off-Surface Rendering.

1. INTRODUCTION

Remote Guidance technologies support distributed collaboration around a physical task and allow expertise in one location to be applied to a task in another location. Typically, this will involve at least one 'worker' who is co-located with the task and one remote 'expert' who will use their expertise to guide the worker through aspects of the task.

Previous research [4, 13, 18] has shown that Remote Guidance technologies are more efficient and effective when they provide

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.

SUI'13, July 20–21, 2013, Los Angeles, California, USA.

Copyright © ACM 978-1-4503-2141-9/13/07...\$15.00.

participants with a "common grounding" [2] upon which they can collaborate. One key aspect of this is some awareness for the worker of the remote expert's view of the workspace.

In many Remote Guidance systems developed to date, the worker has an extremely literal awareness of the remote expert's view of the workspace because that viewpoint is physically embodied by the position of a single streaming video camera, but increasingly Remote Guidance systems are employing depth cameras or other sensors to capture and stream real time 3D representations of the physical workspace [12, 20, 24]. These can be rendered in such a way as to provide the remote expert with the ability to independently manipulate their viewpoint within the workspace. We refer to this process as 'free movement'. As Gutwin et al. observed [7], remote collaboration approaches can obscure people's locations, conceal activities and reduce efficiency when they allow users to dynamically alter their viewpoint of the workspace but do not convey this change to others.

Augmented Reality (AR) can be used to depict the location of a virtual object in free space. In application areas such as telemedicine, maintenance and manual assembly it is often undesirable to encumber the worker's head with an HMD, or hands with a tablet device. Spatial Augmented Reality (SAR) systems which project computer graphics directly into the physical environment have been used to convey guidance to the worker while permitting them to maintain a clear and direct view of the task at hand [1, 11]. While SAR has particular benefits in providing task guidance cues on the surface of physical objects, there has been very little success to date in depicting 3D information that does not exist on or near such a surface.

In this paper, we present three new SAR techniques; *Composite Wedge*, *Vector Boxes*, and *Eyelight*, for visualizing off-surface 3D locations and providing a user with an awareness of the portions of the physical work environment that can be seen from the 3D free movement viewpoint of a remote expert (Figure 1).

In the following sections we review the previous work that pertains to this research problem. We then describe our new techniques and detail our prototype implementations of each of

them. We also present a user study that compares the suitability of each of the techniques in terms of their ability to convey the ‘viewability map’ from an arbitrary floating viewpoint, and their ability to convey the location of an arbitrary off-surface 3D point.

2. RELATED WORK

Gutwin et al. [7] identify and describe four important types of awareness that arise during group work: *Informal Awareness* - a general sense of who's around and what they are up to; *Social Awareness* - information that a person maintains about others in a social context, e.g. their level of interest during a conversation; *Group-Structural Awareness* - knowledge about people's roles and responsibilities in the group; *Workspace Awareness* - maintaining knowledge about other's interactions with the workspace. It is the last of these, Workspace Awareness, which we aim to support with the techniques described in this paper.

In 2D collaborative digital workspaces, previous research has demonstrated a range of techniques for providing Workspace Awareness. The ShrEdit text editing system [3] is an early example of displaying each user's system feedback to all users. Radar Views [7] later showed how small maps representing each user's view of a 2D graphical environment could support Workspace Awareness.

In 3D collaborative virtual environments, Avatars have been often used to represent other users, and even this mimetic representation [10] has been improved by more explicit representations of view frustums and object viewability [9]. In Spatial User Interfaces (SUI), however, visual representations often exist only on the available physical 2D surfaces. Some efforts have been made in previous work to convey task guidance cues to users via SAR [16, 21, 28] but these visual cues use the display surface to convey explicitly authored task goals.

A number of systems have overlaid ‘shadows’ [15, 23] or video [14, 22] of remote participants’ arms and hands directly into the collaborative workspace. The corresponding 3D viewpoint location may be inferred from the assumption they are anatomically human. These systems, however, largely rely on symmetric modes of interaction and become less relevant when the remote worker can choose an arbitrary virtual viewpoint.

Genest et al. [5] presented a technique for representing the height of a finger above a collaborative touch screen. However, their concentric circles visualization was not designed to be linearly mapped to the physical height, nor could it be used within the room surrounding the display surface.

The LightGuide [17] and the LightSpace [27] systems can both display objects at arbitrary 3D locations in a room, however the user must provide their hand or body as a display surface. While these techniques have shown promise in providing gesture guidance and user interaction, they are unsuitable for providing passive visualization of workspace awareness information.

3. VIEWPOINT VISUALIZATION

In this section we describe our three new SAR visualization techniques. These are motivated by application scenarios that include the remote guidance of physical tasks and, as such, we describe the techniques with reference to a table or bench-like display surface, upon which task related objects are placed. A projector then augments the workspace from above. While this arrangement has been used by many researchers to explore visualizations on the table surface, our three new techniques use it to visualize a viewpoint (such as that of a remote expert) that may be effectively ‘floating’ anywhere in the surrounding room.

3.1 Composite Wedge

The Composite Wedge combines a 2D component and a 1D component to convey the 3D location of the target viewpoint (see Figure 2 and Figure 1a). The 2D component specifies the location of the target point on the plane coincident with the projection surface, and is inspired by the ‘Wedge’ off-screen point representation technique developed by Gustafson et al. [6] to convey the 2D location of points positioned outside of a small display screen. The Wedge has the shape of an acute isosceles triangle, with the two base corners of the Wedge rendered on-screen, and the (implied) apex of the triangle coinciding with the target point. The user is able to infer the location of the target point by mentally completing the two lines from the base corners to the apex. Gustafson et al. showed that the Wedge technique can out-perform previous off-screen location techniques in both speed and accuracy, especially in the presence of clutter.

The 1D component of the cue is designed to convey the height of the target point to the user. This is achieved through the use of a relative scale, the bottom and top endpoints map to designated physical objects in the scene (such as the tabletop and the projector/ceiling, respectively). The relative height of the target point between these two extremes is rendered on the scale. The scale endpoints and midpoint are rendered as arrows pointing ‘up’ (in the direction of the upper endpoint, corresponding to the higher reference object) so that the user may easily distinguish the upper and lower endpoints of the scale. When the target viewpoint exists above the display surface, a dot, instead of a wedge is used in combination with this 1D cue.

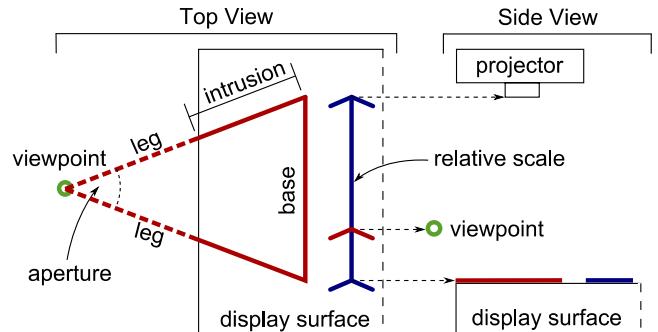


Figure 2. Conceptual diagram of the Composite Wedge cue.

Gustafson et al. describe the main visual components of the wedge as the legs (lines from the base corners to the target point), aperture (angle between the legs), and rotation (the rotation of the Wedge around the target point). Combining these three components give a value called intrusion, defined as the distance of the furthest Wedge base corner from the edge of the screen. Gustafson et al. found that setting intrusion to increase with the distance between the target point and the screen edge gave best results, primarily since it allows more of the legs to be rendered, reducing error for distant target points. They controlled the intrusion through setting the leg length, which is calculated using a non-linear function that levels off for large target point distances. A generalization of Gustafson et al.’s formula for calculating leg length is given by the following equation:

$$leg = dist + \log\left(\frac{dist + A}{B}\right) \times C$$

Where *dist* is the distance from the target point to the display surface, *A* sets the minimum wedge size, *B* controls the rate of change of intrusion relative to *dist*, and *C* linearly adjusts the

intrusion of the Wedge into the display surface. All distances are measured in pixels.

Gustafson et al. provide values $A=20$, $B=12$ and $C=10$. However these values were based on a mobile handset's display resolution and typical target distances to the screen. Instead, we use values $A=300$, $B=200$ and $C=100$ which we have found to be more suitable for tabletop size displays. This sets the minimum wedge size as 40mm, and was also found to achieve a good intrusion-clutter balance.

Gustafson et al. map the aperture of each Wedge to the distance of the target from the screen and leg length. Their original formula was found to be suitable for our purposes, and we thus implement it without change.

A rotation of the Wedge occurs when the Wedge is near a corner of the display surface. Gustafson et al. do not provide details of their corner rotation algorithm, and we therefore developed an original algorithm. This algorithm was designed to satisfy two constraints: visual continuity, and preservation of intrusion. Firstly a spatial buffer equal to the minimum wedge size is defined from each display surface edge. Any points on the display surface and outside the buffer are considered part of the inner display surface. If either of the wedge base points is outside the inner display surface, then the Wedge is rotated until both base points are within the inner display surface, increasing leg length as necessary. This rotation, however, modifies intrusion and violates continuity. To address these issues, we modify the leg length by logarithmically interpolating, based on a rotation degree out of 45 degrees, between the non-rotated wedge and the rotated wedge, resulting in a smooth rotation of the Wedge on corners while maintaining appropriate intrusion.

3.2 Vector Boxes

The Vector Boxes method also uses a form of triangulation to perform 3D viewpoint specification (see Figure 3 and Figure 1b). Two or more vectors, all pointing at a common target, are rendered onto the display surface, such that if the vectors were long enough then the target point would coincide with their intersection. These vectors are placed at different locations (e.g. the corners of a tabletop) on the display surface. This cue was inspired, in part, by the '3D Pathlet' vector used by Sodhi et al. [17] to convey off-surface 3D directions using SAR.

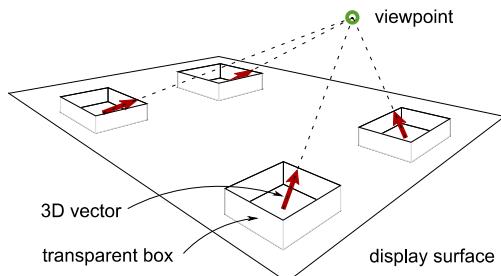


Figure 3. Conceptual diagram of the Vector Boxes cue - the viewer mentally triangulates the vectors to infer the location of the target viewpoint.

Each vector is positioned within a virtual box that appears to be cut-out of the table to the user. This means the arrows appear to the user to be placed below the display surface. By placing the vector below the projection surface we reduce the chance of the vectors apparently being cut off because they exceed the available display surface. Hillinges et al. [8] have previously shown that in a tracked-viewpoint non-stereoscopic system, such as this one,

depth perception can be better when virtual objects appear below or behind the display surface.

3.3 Eyelight

Conceptually, the Eyelight technique works as if a point light has been placed at the physical location of the target point, lighting up the parts of the work environment that are visible from that point, and casting the rest in shadow (see Figure 4 and Figure 1c). In practice this effect is achieved by using a projector to render the lit and shadowed areas into the environment. This cue was inspired by previous work in rendering virtual shadows with SAR, such as the URP system for simulating sunlight conditions for architectural design [26].

Shadows are a part of every person's day-to-day environment and have therefore been explored in previous work as a way to enhance the 'realism' of AR objects. For example, Sugano et al. [19] performed a study using head-mounted video see-through AR, finding that the shadows of virtual objects (a) provide a stronger connection between the real world and virtual objects and so increases virtual object presence, and (b) provide depth cues and so makes three-dimensional perceptions easier for the users of the interface. The converse is also expected to be true; rendering the appropriate virtual shadows within a physical scene would give a good indication of the location of a virtual light source.

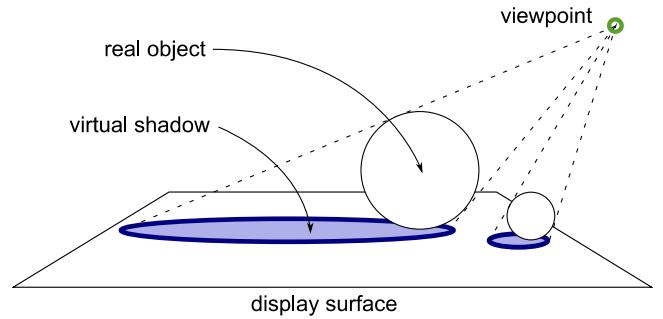


Figure 4. Conceptual diagram of the Eyelight cue - the areas not visible from the target viewpoint are shaded with a virtual shadow.

This cue can be considered to be a quite literal representation of the visibility map of a viewpoint. Areas of the scene which are lit are visible from the light source since by definition they lie on an un-occluded path from the source. Similarly, shadowed areas are not visible from the source.

4. PROTOTYPE IMPLEMENTATION

We constructed a prototype system that implements each of the three techniques described in the previous section. For comparison we also implemented a 'control condition' consisting of a *Screen* showing a simple 'clone' of a real-time 3D view that could be provided to the remote expert. This section describes the main implementation details of the system architecture with calibration process, Vector Boxes, Eyelight and Screen. The Composite Wedge was implemented in OpenGL based on the algorithms described in the previous section.

4.1 System Architecture and Calibration

The hardware utilized in this setup includes a projector with a rigidly attached RGB-D camera (in this case, a Microsoft Kinect) and a tracker system (see Figure 5).

The steps taken to find the pose of each device in a single coordinate system are outlined below:

Projector-Camera: RGBDemo [<http://labs.manctl.com/rgbdemo>] was used to find the relative pose between the camera and the projector.

Tracker-Camera: Tracking markers were attached to a camera shell, which allows the tracker system to sense the position and orientation of a fixed point on the camera. A custom implementation of Hand-Eye calibration [25] was used to determine the transformation between this 6DOF tracked point and the camera's RGB sensor.

Camera-World: This transformation was computed by manually aligning the virtual display surface with a 3D model of the display surface reconstructed from data from the camera's depth sensor. First, a planar model is fitted to the reconstructed 3D model via the Random Sample Consensus (RANSAC) implementation in the PCL [<http://www.pointclouds.org>], thereby extracting the dominant plane. The normal of this plane and the virtual plane are then aligned and manual translation and rotation around the now common normal axis is performed until alignment.

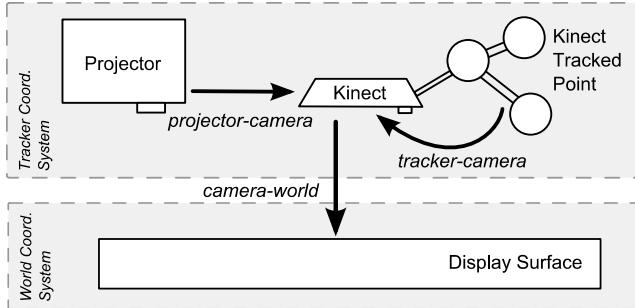


Figure 5. The pair-wise extrinsic relationships computed during calibration. The pose of each device in the world coordinate system can be found by travelling backwards along the arrows from the world coordinate system to the device, and concatenating the poses corresponding to the visited arrows.

4.2 Implementing the Vector Boxes

In this cue, two or more 3D vectors are rendered onto the display surface. Each vector points at the target point, allowing for the user to mentally triangulate the location of this point.

A crucial requirement of this cue is that the vectors appear 3D to the user (Figure 6). In order to achieve perspective-correct rendering, the viewpoint of the user is tracked by having them wear a glasses frame with tracker balls attached (see Figure 7, top right). This allows the approximate location of the user's viewpoint to be streamed from the tracker machine in real time. The streaming was performed using VRPN.

The process for generating the projector-rendered image is now described. A virtual model of the display surface, including the vectors, is rendered from the point of view of the user using a per-frame calculated projection matrix. This projection matrix is computed using the minimum and maximum X, Y and Z values of the display surface corner points transformed to the viewpoint coordinate system, in order to ensure that the display surface fills out the rendered image, thus reducing the aliasing effects that accompany rasterization. The resulting image is temporarily stored in an OpenGL frame buffer object (FBO). The positions of the pixels in the FBO corresponding to the display surface corners are then computed, by applying the projection matrix to each of the 3D corner positions. This information is used to cut out the

portion of the FBO containing the display surface and convert it to a texture. OpenCV was used to perform the perspective warping of this texture, based on the viewpoint of the worker, in order to map it correctly to the virtual display surface. Finally, this texture is applied to the virtual display surface, and rendered onto the physical display surface with the projector.

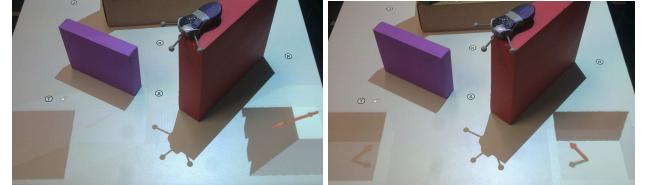


Figure 6. Vector cue viewed from non-user viewpoint in scene (left) and from tracked worker viewpoint (right). In both cases the Vectors are pointing at the 'clicker' on the red box.

Virtual shadows were rendered for each of the virtual vectors and their boxes in this cue, in order to increase the presence of the vectors, and to give the user additional depth cues [19]. The position of the virtual light source used to generate the shadows was selected to be coincident with the physical projector bulb. Shading effects on the vectors were also computed from this light source.

4.3 Implementing the Eyelight

The Eyelight cue functions by 'bathing in light' the portions of a scene which are visible from the target viewpoint and overlaying the rest with 'shadow'. First, a 3D model of the scene is acquired using the Kinect. To minimize the effect of noise in the Kinect sensor, we detected known objects within the obtained point cloud and substituted the respective points with manually created 3D models. Then, the shadows cast within the virtual scene from a given light source are computed using shadow-mapping rendered into the scene with a projector.

Shadow-mapping was implemented using OpenGL, with the aid of the GLEE_ARB extension. Both virtually shadowed and non-shadowed areas within the scene were rendered, in blue and white respectively, in order to distinguish them from real shadows within the scene caused by the projector bulb.

4.4 Control Condition

The non-SAR screen-based technique was also implemented, for comparison. In this method, a virtual camera is placed at the target point, facing the center of the display surface, and the view of the scene from the camera is rendered onto a computer monitor. This essentially shows the same view as a remote expert might see.

The same 3D model that was described in the previous section was rendered to the monitor using OpenGL. In order to aid in distinguishing different objects during the study, each of the real world objects was covered in cardboard of a distinct color, and the corresponding objects in the virtual scene were colored to match.

5. EVALUATIONS

We designed and conducted a user study to compare the relative merits of the new techniques. We investigated both their ability to convey the visibility map for (i.e. what can be seen from) a given point, and also their ability to convey the location in space of a given point.

5.1 Setup

The task environment for the experiment was designed to resemble that of a remote guidance scenario. It involves a table plane where the 'work' is performed, some objects placed on the

table, and various optical devices. A list of the equipment utilized in the task environment is as follows (see Figure 7):

1. rectangular tabletop display surface;
2. six color-coded boxes placed on the display surface;
3. fourteen labeled markers placed on the display surface and boxes, for visibility testing;
4. ceiling mounted projector, to render SAR cues onto the display surface;
5. Kinect, for calibration and model generation; and
6. Optitrack infrared tracker system containing six cameras, to obtain the real-time pose of tracked objects in the scene for interaction and calibration.



Figure 7. (Left) Task environment setup, includes (1) display surface, (2) box objects for occlusion, (3) markers for visibility testing, (4) projector, (5) Kinect, (6) Optitrack tracker system. (Top Right) The tracked glasses frames (Bottom Right) The 3D tracked ‘clicker’.

The cue rendering code was run from a desktop machine with a dual core Intel Xeon E5520 2.26GHz CPU, 3GB RAM, and NVIDIA Quadro FX 4800 GPU, and Windows XP Professional Service Pack 3 (32 bit). Tracking data was obtained and streamed to the rendering machine via Ethernet from a desktop machine with an Intel Core2 Duo E8300 2.83GHz CPU, 2GB RAM, Intel Q35 integrated graphics chipset, and Windows 7 (32 bit).

To facilitate user interaction, a number of tracked objects were utilized in the study. These include a lecture presentation clicker, containing a clickable interaction button, and a pair of glasses, for viewpoint tracking in the Vector Boxes cue. The world coordinate system was defined such that the origin coincided with the center of the display surface, the X and Z axes were parallel to the display surface, and the Y axis was pointing upwards.

5.2 Tasks

5.2.1 Visibility Map Estimation

This task is designed to measure the effectiveness of each cue in conveying a visibility map to the user. During each trial the participant is shown a visual cue, and asked to specify the markers they believe are visible from the viewpoint specified by the cue. User interaction is performed using a pen, a top-down 2D paper diagram of the scene (see Figure 8), and the (for this task, untracked) clicker device. In order to begin a trial, the participant places the clicker over a designated area of the display surface and presses the clicker button to load a viewpoint cue. The participant then circles the corresponding visible markers on the paper diagram. The markers are labeled with letters, both in the scene and on the diagram, in order to ease the mapping between them. The diagram is placed on a clipboard to aid mobility. The

participant presses the clicker again when they are finished, recording the time taken to perform the trial.

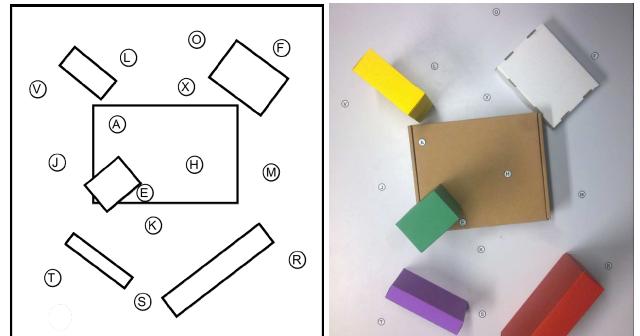


Figure 8. Top-down diagram of the scene provided during the visibility map estimation task (left) and actual top-down view of the scene (right).

5.2.2 Location Estimation

This task is designed to measure the effectiveness of each cue in conveying the 3D location of a viewpoint to a user. For each trial, the participant estimates the location of a target point based on a visual cue rendered in the scene. All user interaction is performed using the tracked clicker device. In order to begin a trial, the participant places the clicker over a designated area of the display surface and presses the clicker button to load a point. This ensures a consistent start position for all trials. The participant is then shown a visual cue corresponding to the trial point, moves the middle ball on the clicker to the 3D position that they believe the cue is conveying, and presses the clicker button once again. On this second button press, the time taken and estimated location are recorded, and the trial is ended.

5.3 Experimental Procedure

A Latin Squares arrangement was used to eliminate ordering effects. With 4 experimental conditions, one for each cue type, 24 participants were recruited to satisfy this arrangement, with each participant experiencing the conditions in a different experiment order. For each participant, the cues, tasks and background were described by the experimenter. The system was then run in ‘interactive mode’, where the target point is locked to the clicker’s position in the physical world, allowing the participant to move the target point around and observe the behavior of the projected cues.

Then, for each of the conditions in the experiment, the participant:

- Is again briefed on the cue corresponding to the condition.
- Performs 5 location estimation practice trials.
- Performs 20 location estimation experimental trials.
- Performs 1 visibility map estimation practice trial.
- Performs 5 visibility map estimation experimental trials.

Every participant was asked to perform each task as quickly and accurately as possible. No feedback on the correctness of guesses made by participants was given during the practice or experimental trials for either task, since no direct feedback would be given in a real world application of this system.

In the location estimation task, a set of 20 fixed target test points, presented in different orderings, were used for all conditions over all participants. The number of points was chosen to be high enough to make it unlikely that participants would notice the same set of points being used each time.

In the visibility map estimation task, due to the much lower number of test trials, it was reasoned that participants would likely notice the re-use of the same five target points across all conditions. Thus

four different sets of five target points, all having a similar spread, were created, and each participant experienced these four sets in the same ordering. Due to the Latin Squares arrangement, each condition was applied to each set an equal number of times.

A brief post-experiment survey was performed. This survey asked users to rate their own performance in terms of time and accuracy for each of the tasks.

5.4 Data Collection

Two major quantities were measured during each trial: the cost, which is inversely proportional to accuracy, of the guessed result, and the time taken to complete the task (to millisecond precision). The cost for the point location task is defined as the Euclidean distance between the guessed point and the ground truth, while the cost of the visibility map estimation task is defined as the number of letters labeled incorrectly.

6. RESULTS

6.1 Visibility Estimation Performance

The results of this task indicate the effectiveness of each cue type, in terms of time and cost, at conveying the visibility map of a viewpoint. Timing comes from the manual timer controlled by the participant using the clicker. Cost was measured as the number of incorrect visibility labelings.

6.1.1 Task Completion Time

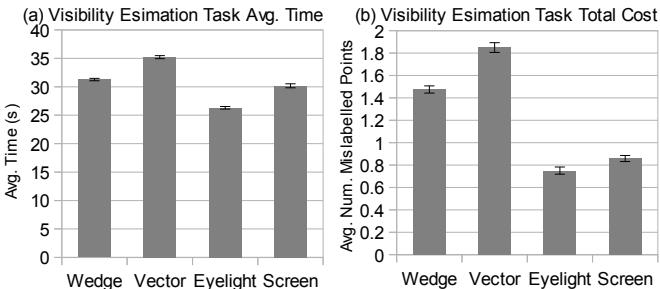
The Eyelight cue was found to have the shortest completion time for this task, with a mean of 26.30 (SD 9.70) seconds. The Vector Boxes cue appeared to have the longest completion time, with a mean of 35.21 (SD 12.04) seconds (see Figure 9a).

A two-way (condition \times trial) repeated measures ANOVA was applied to the time data after a log transformation had been applied. This transform was used to satisfy the assumption of homogeneity in the data. Completion times differed significantly across the four types ($F(3,449)=23.51$, $p<.001$). The analysis also showed that trial number affected completion time ($F(4,449)=6.36$, $p<.001$).

A post-hoc Scheffe test (significance level 0.05) was run on the data, and a significant difference was found between the two fastest cues (Eyelight and Screen) and the two slowest cues (Composite Wedge and Vector Boxes).

6.1.2 Task Cost

Figure 9b shows the total task cost for the visibility map estimation task over all participants and trials for each condition. The Eyelight cue had the lowest number of mislabeled points, followed closely by the Screen cue. The Composite Wedge and Vector Boxes cues had a higher cost, with almost twice the mislabel rate of the previous cues. A Wilcoxon signed-rank test showed all cost differences were significant ($p<0.05$).



6.2 Location Estimation Performance

6.2.1 Task Completion Time

The Composite Wedge cue was found to have the lowest completion time, with a mean of 6.560 (SD 2.671) seconds. The task with the longest completion time is the Screen cue, with a mean of 12.664 (SD 6.789) seconds (see Figure 9c).

A two-way (condition \times trial) repeated measures ANOVA was applied to the (also log transformed) time data to test for differences in completion time among the four cue types. Completion times differed significantly across the four types ($F(3,1873)=206.87$, $p<.001$). The analysis also showed that trial number had a minor effect on completion time ($F(19,1873)=8.02$, $p<.001$). Note that there was one outlier in the data, omitted from the calculations, caused by the user accidentally skipping a trial by clicking in quick succession.

Performing the Scheffe post-hoc test (significance level 0.05) on the data revealed that a significant difference exists between the mean of the fastest cue (Composite Wedge) and each of the other cues. Additionally, a significant difference was found to exist between the slowest cue (Screen) and all the other cues.

6.2.2 Task Cost

The Composite Wedge cue had the lowest estimation cost, with a mean of 207.4 (SD 187.2) mm. The Screen cue had the highest cost for this task, with a mean cost of 379.7 (SD 247.6) mm (see Figure 9d).

A two-way (condition \times trial) repeated measures ANOVA was used to find that there was a significant difference in task cost between conditions ($F(3,1873)=119.24$, $p<.001$). Additionally, trial number had an effect on task cost ($F(19,1873)=29.22$, $p<.001$). This analysis also omits the outlier mentioned in the previous section.

Applying the Scheffe post-hoc test (significance level 0.05) revealed that there was a significant difference in cost between the most accurate cue (Composite Wedge) and the other cues. The Vector Boxes cue was also found to have a significantly lower cost than the Eyelight and Screen cues.

6.3 Questionnaire

6.3.1 Visibility Estimation

Most participants felt that they performed the visibility map estimation task more quickly using the Eyelight and Screen cues than with the Vector Boxes and Composite Wedge cues (see Figure 10). The majority of participants asserted that, even when using the latter two cues, they performed the task with an above average accuracy.

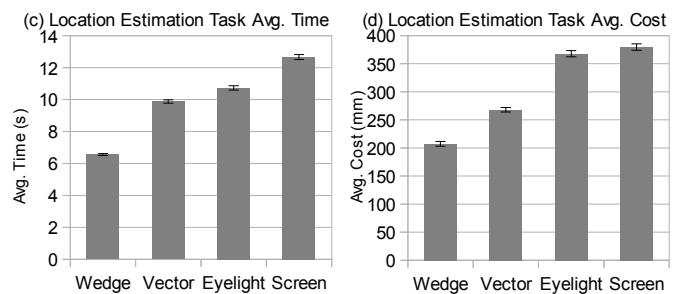


Figure 9. Results for Times and Costs for the Visibility Estimation and Location Estimation tasks. Bars indicate standard error.

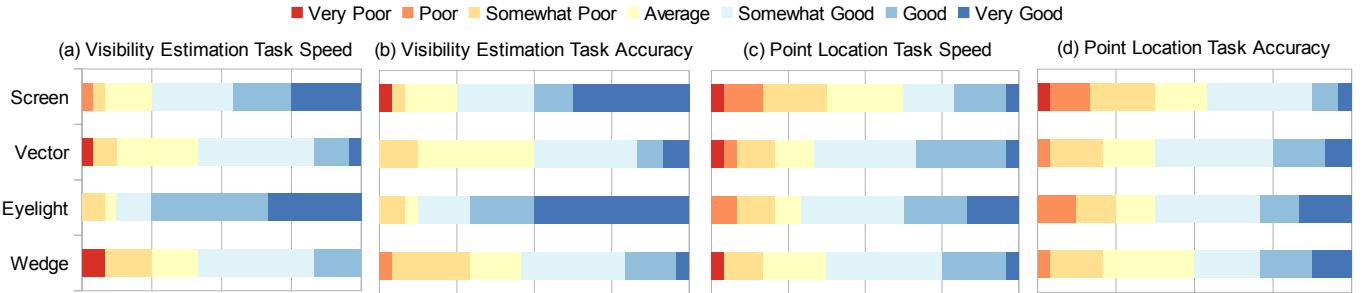


Figure 10. Participant responses to the qualitative survey.

Similarly, the majority of participants perceived that they performed the visibility map estimation task more accurately with the Eyelight and Screen cues than with the Composite Wedge and Vector Boxes cues. Half of the participants felt that the Eyelight cue enabled them to complete the task with the highest possible accuracy.

6.3.2 Location Estimation

The Eyelight, Vector Boxes and Composite Wedge cues had similar perceived task completion times. Participants generally thought they performed slower with the screen cue than the other cues.

Most of the participants believed that they completed the point location task with an above average accuracy using the Eyelight and the Vector Boxes cue. The Screen cue has the lowest overall perceived completion accuracy for this task.

7. DISCUSSION

The results of our study are particularly encouraging and indicate that SAR can indeed be used to visualize off-surface information. No single approach was found to perform better for conveying **both** a visibility map and a point's 3D location: the Composite Wedge is a 'clear winner' in terms of 3D location estimation, while the Eyelight and the Screen cues performed best for the visibility map estimation task. We note that a system could, for example, simultaneously use an Eyelight for workspace awareness and reinforce the visualization with the Composite Wedge depicting the viewpoint's location.

The Vector Boxes cue did not come out on top in either the visibility estimation or the location estimation tasks. We expect this is, in part, because the large separation between arrows needed to reduce uncertainty in triangulation resulted in more 'looking around' in the workspace for the worker.

In the location estimation task, the distance of the target point from the centre of the display surface appears to have an effect on cost, with closer points having a higher mean accuracy. This is unsurprising, since as distance increases, the performance of every cue is expected to degrade. Even the Screen technique shows a smaller image on the scene when the distance is greater.

In the location estimation task, the amount of error in the Y (up) dimension tended to be much greater than that of either the X or Z dimensions for all but the Screen cue, which had a similarly high error in all three dimensions (see Figure 11).

Over half the participants rated themselves as performing the location estimation task with greater than average accuracy using the Eyelight cue, despite having the third highest mean cost during the task. This suggests that the Eyelight cue entailed a false sense of confidence, and in this sense may even be slightly

misleading when no feedback is given. This may stem from the fact that participants frequently experience shadows in the real world, with this constant exposure leading them to believe they are able to perform the task with a high accuracy.

Participants generally perceived themselves as taking longer to perform the location estimation task with the Screen cue than with the other cues. It was noted that numerous participants verbally commented on this fact during the study. One participant commented that the difficulty of this cue was due to a lack of visual feedback for this cue when moving around, since it is not projected into the physical workspace like the others.

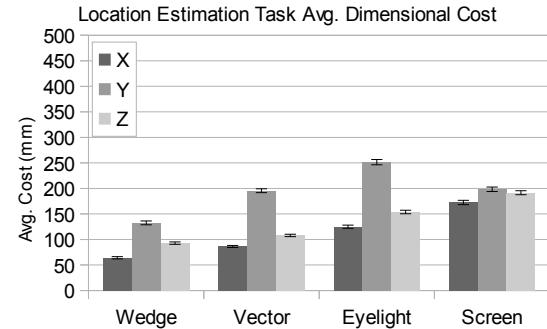


Figure 11. The average cost over all participants in the X, Y and Z dimensions for the location estimation task. Error bars indicate standard error.

8. CONCLUSION AND FUTURE WORK

We have presented the design, implementation and evaluation of three novel techniques of providing viewpoint awareness using SAR visual cues. The user study showed that the Composite Wedge cue was the best for providing location awareness, and the Eyelight cue was the best for providing visibility map awareness. These two cues do not appear to be conflicting or incompatible, so when both types of awareness are likely to be useful, a combination of the Composite Wedge and Eyelight cues should be considered.

The most obvious extension to this work is to consider multiple remote viewers, and their associated viewpoints. Encouragingly, the 2D Wedge has previously been shown to work with multiple instances and we believe a handful of Eyelights, each with different colors, may also prove effective. Future research could also address augmenting the cues presented here with information about the field of view of the viewpoint, as suggested in [9]. There is also scope to reduce the Y-axis (up) errors through modifications to these visualization techniques.

Further studies could more thoroughly investigate the behavior of each of our techniques when conveying a dynamic viewpoint, or when the user's attention is partially invested in a particular task.

There is also much scope to investigate the performance of these visualizations on non-planar surfaces.

9. ACKNOWLEDGMENTS

We sincerely thank: the user study participants, Warren Muller for his expert statistics advice, John Zic and Chris Gunn for their support, and the SUI'13 anonymous reviewers for their comments and feedback.

10. REFERENCES

- [1] Adcock, M. and Gunn, C. 2010. Annotating with “sticky” light for remote guidance. *ACM SIGGRAPH ASIA 2010 Posters* (Seoul, South Korea, 2010).
- [2] Clark, H.H. and Brennan, S.E. 1991. Grounding in communication. *Perspectives on socially shared cognition*. American Psychological Association.
- [3] Dourish, P. and Bellotti, V. 1992. Awareness and coordination in shared workspaces. *Proceedings of the 1992 ACM conference on Computer-supported cooperative work - CSCW '92* (Toronto, Canada, Dec. 1992), 107–114.
- [4] Fussell, S., Setlock, L., Yang, J., Ou, J., Mauer, E. and Kramer, A. 2004. Gestures Over Video Streams to Support Remote Collaboration on Physical Tasks. *Human-Computer Interaction*. 19, 3 (Sep. 2004), 273–309.
- [5] Genest, A. and Gutwin, C. 2012. Evaluating the effectiveness of height visualizations for improving gestural communication at distributed tabletops. *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12* (Seattle, USA, Feb. 2012), 519.
- [6] Gustafson, S., Baudisch, P., Gutwin, C. and Irani, P. 2008. Wedge: clutter-free visualization of off-screen locations. *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08* (Florence, Italy, Apr. 2008).
- [7] Gutwin, C., Greenberg, S. and Roseman, M. 1996. Workspace Awareness in Real-Time Distributed Groupware: Framework, Widgets, and Evaluation. (Jan. 1996), 281–298.
- [8] Hilliges, O., Kim, D., Izadi, S., Weiss, M. and Wilson, A. 2012. HoloDesk: Direct 3D Interactions with a Situated See-Through Display. *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12* (Austin, USA, May. 2012), 2421.
- [9] Hindmarsh, J., Fraser, M., Heath, C., Benford, S. and Greenhalgh, C. 1998. Fragmented interaction: establishing mutual orientation in virtual environments. *Proceedings of the 1998 ACM conference on Computer supported cooperative work - CSCW '98* (Seattle, USA, Nov. 1998), 217–226.
- [10] Hutchins, M., Adcock, M., Stevenson, D., Gunn, C. and Krumpholz, A. 2005. The Design of Perceptual Representations for Practical Networked Multimodal Virtual Training Environments. *Proc. HCI International 2005: 11th International Conference on Human-Computer Interaction* (2005).
- [11] Iyoda, T., Abe, T., Tokai, K., Sakamoto, S., Shingu, J., Onuki, H., Shi, M., Uchihashi, S., Satoh, S., Nack, F. and Etoh, M. 2008. LightCollabo: Distant Collaboration Support System for Manufacturers. *Advances in Multimedia Modeling* (Berlin, Heidelberg, 2008), 369–379.
- [12] Kainz, B., Schmalstieg, D., Hauswiesner, S., Reitmayr, G., Steinberger, M., Grasset, R., Gruber, L., Veas, E., Kalkofen, D. and Seichter, H. 2012. OmniKinect: real-time dense volumetric data acquisition and applications. *Proceedings of the 18th ACM symposium on Virtual reality software and technology - VRST '12* (Toronto, Canada, Dec. 2012), 25.
- [13] Kirk, D. and Stanton Fraser, D. 2006. Comparing remote gesture technologies for supporting collaborative physical tasks. *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06*. (2006), 1191.
- [14] Kirk, D.S. 2006. *Turn It This Way!: Remote Gesturing in Video-Mediated Communication David Stanley Kirk , BSc (Hons) , MSc Thesis submitted to the University of Nottingham for the degree of Doctor of Philosophy*. University of Nottingham.
- [15] Robinson, P. and Tuddenham, P. 2007. Distributed Tabletops: Supporting Remote and Mixed-Presence Tabletop Collaboration. *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '07)* (Oct. 2007), 19–26.
- [16] Rosenthal, S., Kane, S.K., Wobbrock, J.O. and Avrahami, D. 2010. Augmenting on-screen instructions with micro-projected guides: When it Works, and When it Fails. *Proceedings of the 12th ACM international conference on Ubiquitous computing - Ubicomp '10* (Copenhagen, Denmark, Sep. 2010), 203.
- [17] Sodhi, R., Benko, H. and Wilson, A. 2012. LightGuide: projected visualizations for hand movement guidance. *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12* (Austin, USA, May. 2012), 179.
- [18] Stevenson, D., Li, J., Smith, J. and Hutchins, M. 2008. A collaborative guidance case study. *AUIC '08* (2008).
- [19] Sugano, N., Kato, H. and Tachibana, K. 2003. The effects of shadow representation of virtual objects in augmented reality. *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.* (2003), 76–83.
- [20] Söderholm, H.M., Sonnenwald, D.H., Cairns, B., Manning, J.E., Welch, G.F. and Fuchs, H. 2007. The potential impact of 3d telepresence technology on task performance in emergency trauma care. *Proceedings of the 2007 international ACM conference on Conference on supporting group work - GROUP '07* (Sanibel Island, USA, Nov. 2007), 79.
- [21] TSIMERIS, J.A. 2010. *Visual Cues for the Instructed Arrangement of Physical Objects Using Spatial Augmented Reality (SAR)*. University of South Australia.
- [22] Tang, A., Neustaedter, C. and Greenberg, S. 2006. VideoArms: Embodiments for Mixed Presence Groupware. *The 20th BCS HCI Group Conference* (London, 2006).
- [23] Tang, J.C. and Minneman, S. 1991. VideoWhiteboard: Video Shadows to Support Remote Collaboration. *Proceedings of the SIGCHI conference on Human factors in computing systems Reaching through technology - CHI '91* (New Orleans, USA, Mar. 1991), 315–322.
- [24] Tecchia, F., Alem, L. and Huang, W. 2012. 3D helping hands: a gesture based MR system for remote collaboration. *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry - VRCAI '12* (Singapore, Dec. 2012), 323.
- [25] Tsai, R.Y. and Lenz, R.K. 1989. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*. 5, 3 (Jun. 1989), 345–358.
- [26] Underkoffler, J. and Ishii, H. 1999. Urp: a luminous-tangible workbench for urban planning and design. *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99* (Pittsburgh, USA, May. 1999), 386–393.
- [27] Wilson, A.D. and Benko, H. 2010. Combining multiple depth cameras and projectors for interactions on, above and between surfaces. *Proceedings of the 23rd annual ACM symposium on User interface software and technology - UIST '10* (New York, New York, USA, Oct. 2010), 273.
- [28] Zhou, J., Lee, I., Thomas, B., Menassa, R., Farrant, A. and Sansome, A. 2011. Applying spatial augmented reality to facilitate in-situ support for automotive spot welding inspection. *Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry - VRCAI '11* (Hong Kong, Dec. 2011), 195.