Virtual displays for 360-degree video

Stephen Gilbert*, Wutthigrai Boonsuk, and Jonathan W. Kelly
Iowa State University, 1620 Howe Hall, Ames, IA USA 50011-2274

ABSTRACT

In this paper we describe a novel approach for comparing users' spatial cognition when using different depictions of 360-degree video on a traditional 2D display. By using virtual cameras within a game engine and texture mapping of these camera feeds to an arbitrary shape, we were able to offer users a 360-degree interface composed of four 90-degree views, two 180-degree views, or one 360-degree view of the same interactive environment. An example experiment is described using these interfaces. This technique for creating alternative displays of wide-angle video facilitates the exploration of how compressed or fish-eye distortions affect spatial perception of the environment and can benefit the creation of interfaces for surveillance and remote system teleoperation.

Keywords: surveillance, remote sensing, tiled display, virtual environment, texture mapping, 3D graphics, wearable computing

1. INTRODUCTION

Recent years have yielded a growing number of 2D visual representations of 360-degree visual data. Google Street View[1, 2] and Microsoft Photosynth[3] offer tiled imagery in a circular configuration. GoPano.com and its Micro camera offer 3D video recording and playback via a traditional looking video player that allows panning to see the full 360 degrees. All of these displays offer access to the full circle of imagery, but show only an excerpt at any given time. Other systems have used a wide variety of display configurations to display all 360 degrees simultaneously on a traditional 2D display.[4-6] It is not clear, however, how best to display the 360 degrees to maximize appropriate spatial cognition and awareness of the full space. The authors conducted a more recent study to compare the effectiveness of three such interfaces.[7] The current paper describes an innovative technique used to create the three interfaces with tiled virtual cameras within a 3D game engine.

Before considering virtual video cameras, we review what has been done to display 360-degree real video. The simplest method can be achieved by combining video feeds from multiple cameras with limited fields of view (FOV) to obtain a wider FOV. However, producing a continuous 360-degree view using this method remains a challenge. Several studies have attempted to address this challenge by proposing techniques to combine and register video feeds, including Image Blending,[8] Piecewise Image Stitching,[9, 10] and 2D Projective Transformation 11, 12. Image Blending uses a weighted average for blending image edges without degrading image details at the border. Park & Myungseok 13 used this technique to combine video feeds from multiple network cameras for developing a panoramic surveillance system. Piecewise Image Stitching computes the correct lens distortion and mapped multiple images onto a single image plane. This technique combines video images from multiple adjacent cameras for a teleconference system called FlyCam 9. Projective Transformation uses the development of a video-based system called immersive cockpit 14.

Our system simulates video feeds within a game engine virtual environment and utilizes multiple virtual cameras to produce a 360-degree view. The technique to combine video feeds from multiple cameras is similar to the Piecewise Image Stitching technique. However, since the FOV of virtual camera is easily adjustable, combining multiple cameras with a small FOV (25-30 degrees) can produce a pleasant 360-degree view without the need for complex image registration.

* gilbert@iastate.edu; 515-294-6782; www.vrac.iastate.edu/~gilbert

Human Vision and Electronic Imaging XVII, edited by Bernice E. Rogowitz, Thrasyvoulos N. Pappas, Huib de Ridder, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 8291, 82911L • © 2012 SPIE-IS&T
CCC code: 0277-786X/12/$18 • doi: 10.1117/12.912173

1.1 Background: The Homunculus Project

This effort is part of a U.S. Army-funded Advanced Live, Virtual, and Constructive training project, which is an effort to address the U.S. Army's requirements for the research and development needed to create the Army's Common Virtual Environment (CVE) for training. In particular, this research stems from a vision of the warfighter as an intelligent data-collecting probe in the field. The fighter wears a lightweight 360° camera and microphone (or set of cameras), along with possibly other sensors, and this information is conveyed back to a Home Station and optionally to additional commanders. The 360° video can be used for two primary objectives:

1. Enable a commander, squad member, or automated agent to monitor the full experience of a warfighter in the field, potentially supplementing that fighter's situational awareness ("There's a sniper behind you at 5 o'clock").
2. Automatically update geospatial databases with intelligence, imagery, and 3D models of buildings using computer vision algorithms on the imagery generated by the warfighter's movement within an urban environment[15], somewhat like a human Google StreetView vehicle[2].

In this paper we focus on the first objective and the challenge of designing an appropriate software interface for the commander or other squad members to experience another individual's 360° video (and potentially audio), as well as multiple individuals' video simultaneously. This surveillance interface shares some goals with a traditional security surveillance system with an array of multiple video signals, but we are particularly interested in whether viewers of the videos can gain accurate spatial awareness of the settings displayed. Czerwinski, et al[16] note the importance of a wide field of view for spatial tasks, and it's possible that compression of that wide field of view to a smaller physical space would aid in performance within a limited display context.

This project is dubbed "Homunculus," derived from the term referring to a "little man" inside the head, an idea considered by early neuroscientists. The 360-wearable camera system, in effect, can place an observer inside the head of the warfighter, able to co-experience life with the fighter. The camera system worn in the field is called the HomCam.

1.2 Other Applications of the HomCam

While the HomCam was originally designed for surveillance, its ability to offer telepresence within a remote area via an observer in the field is a valuable contribution in non-military domains. In construction firms, builders would like to be able to monitor building progress remotely, and construction engineering departments would like to put a HomCam-style rig on engineers in the field to provide students with virtual field trips that would otherwise be cost prohibitive.[17] Simple mobile videoconferencing has been widely discussed for virtual field trips for students (Bergin REF) such as factory tours[18] and museums[19] since Wi-Fi bandwidth began to support mobile video transmission. Mobile videoconferencing has also been used for teleconsultation within hospitals[20] and for computer technicians in the field.[21] Previous work in this direction has been done in the context of remote sensing and wearable computing. The HomCam distinguishes itself by offering remote access to 360-degree video.

2. APPROACH

To understand how a 360-degree view can influence people's spatial perception and performance, we designed a software environment for conducting experiments to investigate the effectiveness of various designs of 360-degree view interfaces on spatial tasks, including exploring, searching, and identifying locations and directions of particular objects (targets). The design of the system is described below and offers other researchers an approach for exploring arbitrarily shaped displays of video and virtual video.

The design of the interface for Homunculus observation and interaction could be developed independently of the hardware using virtual environments by creating a virtual entity which possessed the Homunculus camera system.

2.1 Simulated video via multiple cameras

Typical game engines allow programmers to create views from multiple camera viewpoints using two different techniques: 1) multiple viewports and 2) render-to-texture. Multiple viewports is a method to display a projection of each virtual camera in each individual viewport. Render-to-texture is a method to render camera views as an image texture that can be mapped onto an arbitrary shape. It is typically used to place camera views on a surface such as an in-game monitor or to create a mirror effect. This method can be used in either single viewport or multiple viewports. In our pre-development of the 360-degree view system, the render-to-texture method provided several benefits that were essential

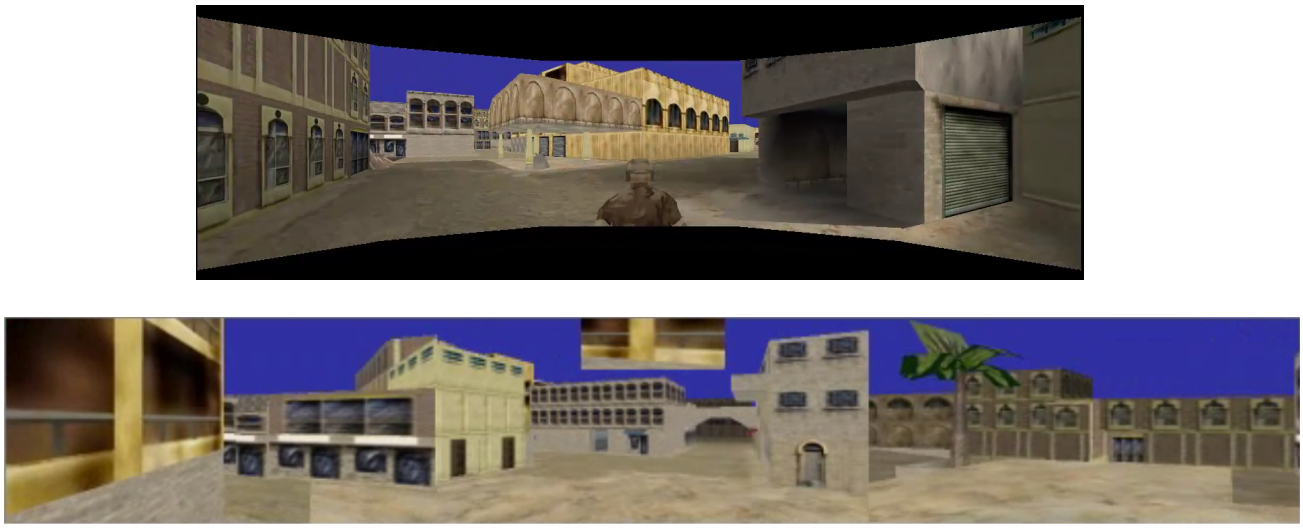


Figure 1: Example layouts created using the texture approach. The "bow tie" layout has a FOV of 225 degrees with five 45-degree cameras. The "rear view mirror" layout has 270 FOV (three 90-degree cameras) in its primary view, with a fourth 90-degree camera inset at top center with normal reversed as a rear-view mirror.

for research studies. One benefit is that the programmer can treat the video texture as one of the basic entities that are allowed basic 3D transformations such as translation, rotation, and scale in the 3D environment. Another benefit is that the render-to-texture method allows programmers to easily manipulate the shape of the texture view, which is useful for exploring alternative designs of a 360-degree view interface. Some initial alternatives for wide-angle / panoramic video observation that have been explored using Delta3D and Irrlicht video game engines are shown in Figure 1.

The 360-degree view was created by combining views of multiple virtual cameras circularly arranged on the same horizontal level in the virtual environment. The view interface was developed using render-to-texture method. A group of moving cameras and one fixed camera were used. The group of moving cameras captured 360-degree view from specific location in the virtual environment. The view from each camera was rendered as texture and subsequently mapped onto rectangular surface arranged and positioned outside a 3D scene. The fixed camera displayed these texture views on the monitor screen. Figure 2 illustrates how the group of moving cameras and the fixed camera are utilized.

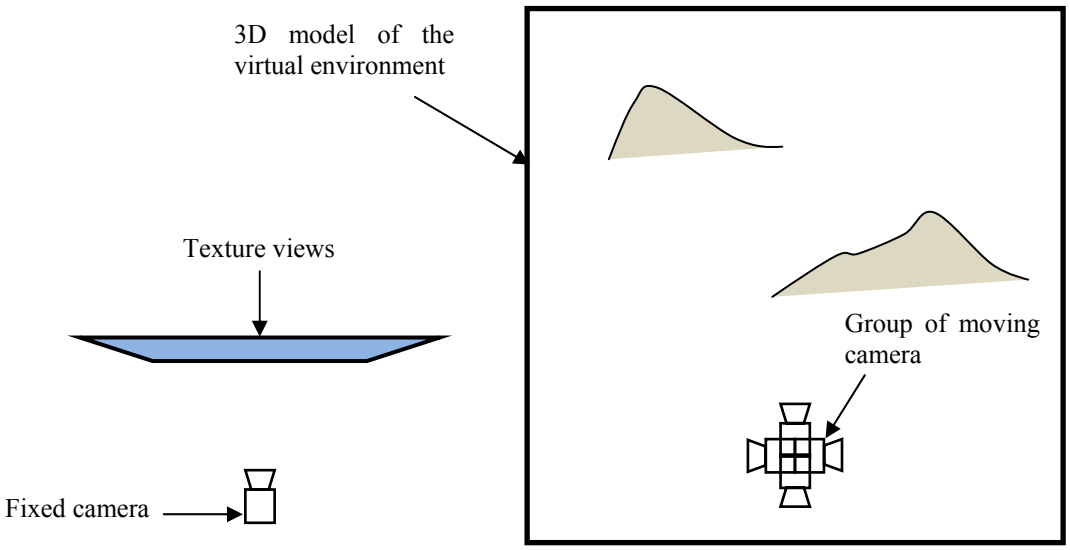
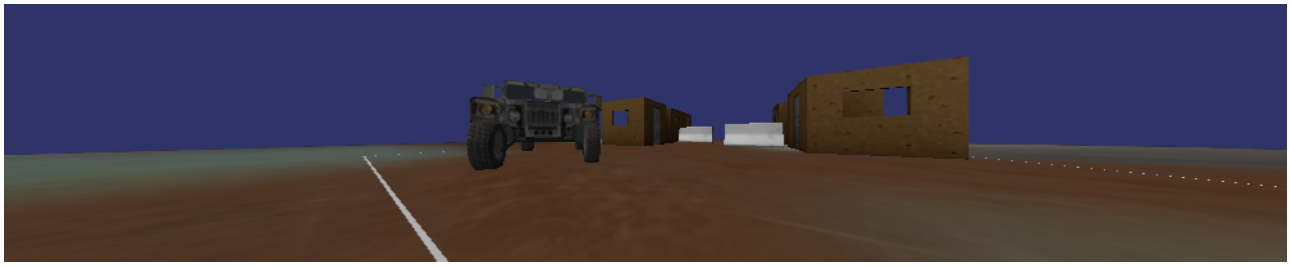


Figure 2. Fixed camera (left) and group of moving cameras (right) in the 360-degree view. The group could have an arbitrary number of cameras arranged radially.

Wide-angle 90-degree FOV

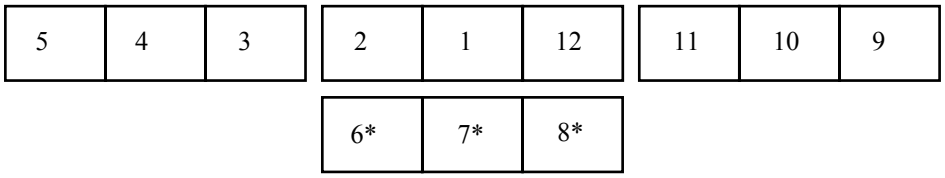
30x3: 90 degree FOV based on three 30-degree FOV tiles



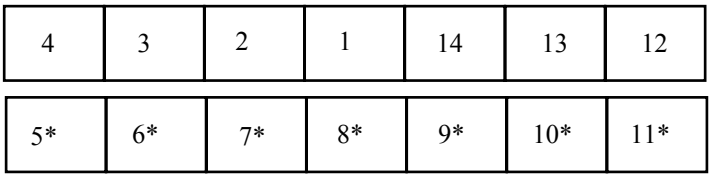
Wide-angle 150-degree FOV

30x5: 150-degree FOV based on five 30-degree FOV tiles

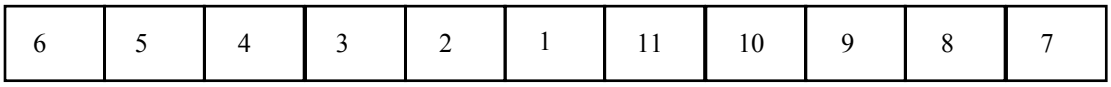
Figure 3: Wide-angle FOV displays vs. tiled displays.



(a) 90-degree x 4



(b) 180-degree x 2



(c) 360-degree x 1

Figure 4. Virtual camera view arrangements; The cameras with * were rendered on surfaces with their normal directions reversed to create a rear-view mirror effect.

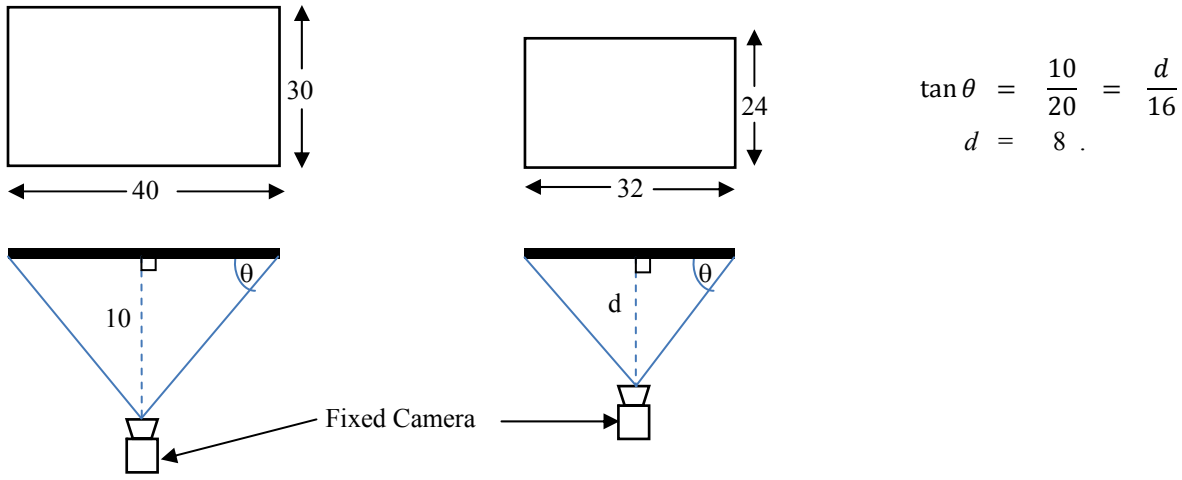
The game engines typically limit a camera with FOV larger than 180 degrees; thus, at least two cameras are required to display 360-degree view. Moreover, as illustrated in Figure 3, distortion of the video image increases significantly when FOV of the camera is ranged from 90 degrees to near 180 degrees. Objects near the center move further away with increasing fish-eye distortion. The tiled displays provide a view in which each individual object appears more natural and close by, though they also create view-wide peculiarities such as mixed perspective vanishing points (see particularly the walls in the 5x30 view in Figure 3). The experiment described below and other research is needed to ascertain the perceptual effects of such peculiarities.

For the experiment, we used this approach to compare three 2D interfaces for 360-degree video that were inspired by others. Kadous et al[5] used a 90-degree x 4 approach for robot teleoperation with separate views for each 90-degree camera (front, left, right, back). Meguro et al[6] used a 180-degree x 2 approach (front and rear) for mobile surveillance. Greenhill and Ventakesh[4] used panoramic 360-degree x 1 approach for surveillance from city buses. Three interfaces with these characteristics were created by tiling virtual cameras with small FOV (~25 to 30 degrees) as shown in Figure 4 with 12, 14, and 11 moving cameras, respectively.

In each interface, the moving cameras were radially rotated at equal angles. One camera was assigned as a front camera that can move as the user manipulates it and the remaining cameras followed the same transformation of this camera. The texture views of the moving cameras were arranged as shown in Figure 2. Camera #1 was set as a front camera and the camera numbers were incremented in counter clockwise order.

The scene aspect ratio (width x height) was set to 4:3. Since the number of moving cameras in each interface was different, the texture view for each interface had different dimensions that resulted in unequal size of objects in the virtual environment when they were displayed on the monitor screen. Therefore, the distance between the fixed camera and the texture view had to be adjusted so that an object was similarly sized in all interfaces. Figure 5 illustrates an example of how the distance between fixed cameras and the interfaces can be adjusted. Numbers in the following example are used for demonstration purposes only and are not the actual numbers for creating the 360-degree view interfaces.

In this example, there are two texture views with dimensions of 40 x 30 units and 32 x 24 units (width x height), respectively. The fixed camera in the first setting has a distance of 10 units from the texture view. In the second setting, distance (d) from the fixed camera to the texture view needs adjusted so the texture views for both settings appear the same size on the monitor screen. The equation below shows d is computed by using $\tan\theta$ for the first setting. The results show the fixed camera of the second setting needs to be closer to the texture view (8 units).



$$\tan\theta = \frac{10}{20} = \frac{d}{16}$$
$$d = 8 .$$

Figure 5. Distance adjustment from the fixed camera can be adjusted as shown to ensure equal sizing of the same object within different size views.

2.2 Collision detection on texture views

In our experiment, a touch screen was used for allowing users to select objects in the scene. The touch point was passed to the game engine as a mouse click. Typical game engines use ray intersection to pick objects in the 3D scene from a 2D screen location. A ray is first generated from the screen coordinate of a picking point. Then, the collision (intersection) between this ray and the object in the view is identified. However, this method only works when camera views are directly rendered on the monitor screen (i.e., multiple viewport rendering method). In the virtual camera approach, images from the moving cameras are rendered on the textures and displayed by the fixed camera to the monitor screen. If the traditional ray intersection method were used, choosing a picking point on the screen would result in only selecting the texture view objects. To solve this problem, the collision detection on objects within the texture views is simulated by connecting a picking point on the texture view to a ray that casts from its respective virtual camera.

Figure 6 illustrates the process of collision detection on the texture views. When a user taps on the monitor screen, a ray is generated from this picking point, based on the screen coordinates. If the ray intersects with one of the texture views, it means the view of that moving camera is selected. This intersecting point is transformed to the view coordinate of the selected moving camera. A new ray is computed using this view coordinate and finally intersection with objects in the scene is identified.

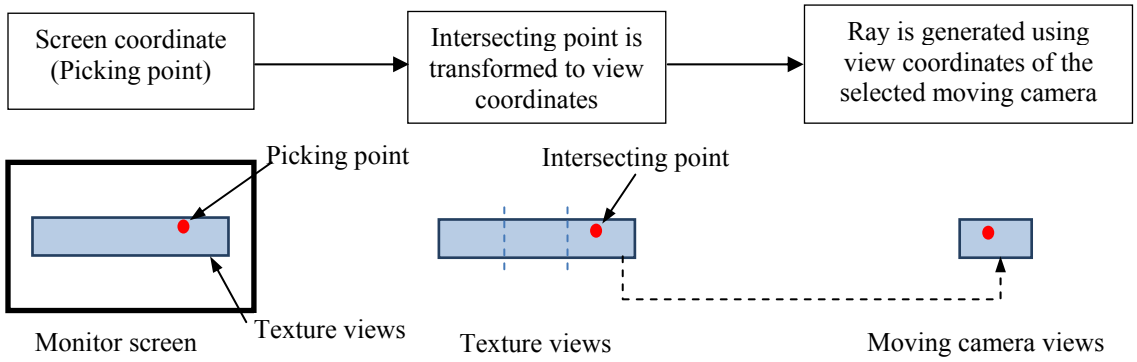


Figure 6. Process of collision detection on texture views

3. EXPERIMENT

Three different 360-degree interfaces were created using the proposed method: 1) 90-degree x 4, 2) 180-degree x 2, and 3) 360-degree x 1 (Figure 7), inspired by existing interfaces for 360-degree views. Twenty participants were recruited to participate in the study. Each participant utilized all three interfaces in counterbalanced order to perform two consecutive tasks. At the beginning of an interface session, participants had up to 5 minutes to learn to control the navigation and get familiar with the view interface using a different 3D environment than the one used during testing. For the first task, participants needed to locate 10 targets (red barrels) within 10 minutes. Participants were instructed to tap on the target as soon as it appeared anywhere on the display. Immediately after selecting the barrel target, participants needed to identify the direction of the target relative to their heading direction on a compass rose (Figure 8). At the end of each session with a given interface, after locating all targets or 10 minutes had passed, participants were asked to locate the targets on the overhead map as shown in Figure 9.

Experimental results showed that participants made significantly fewer errors when identifying the angle of targets in relation to themselves (an egocentric, or first-person view) when using the 90-degree x 4 and the 180-degree x 2 interfaces compared to the 360-degree x 1 interface (the first two interfaces yielded no significant difference). This result may stem from their being less distortion in the 90-degree x 4 and 180-degree x 2 interfaces than in the 360-degree x 1 interface. Also, the center window of 90-degree x 4 interface and the top window of 180-degree x 2 interface provide front views that are more comparable to the human natural front view while the back views are displayed in separate windows. The effect also may be due to the visual boundaries that exist in those two interfaces. In the 90-degree x 4, for example, the edges of the front view demark -45 and +45 degrees.

The results of the map task explored whether participants could translate an egocentric view to a top-down map view

(exocentric). However, the difference of the interface design did not influence participants' abilities to recall the locations of targets in the virtual environment (there were no significant differences in map errors by interface). Also, across all three interfaces, the performance of the pointing task also was not correlated to the performance of the map task. The differential effects of interface on egocentric pointing and map placement suggest these two tasks might rely on psychologically independent processes. More detailed results can be found in Boonsuk et al.[7]

Figure 7. Three 360-degree interfaces from the same viewpoint: (a) 90-degree x 4, with left, front, right, rear; (b) 180-degree x 2, with front and rear and (c) 360-degree x 1, panorama.

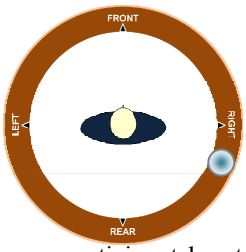


Figure 8. Compass rose, where a participant has tapped to indicate that a target barrel sits about 100° to the right of her heading.

Figure 9. Overhead map of terrain, on which participants located the target barrels that they had selected in first-person view.

4. DISCUSSION

Significant research has been done on how people navigate in virtual environments.[22-25] It is likely, though must be confirmed, that these results also apply to navigating real environments via video using teleoperation. Primary research questions include: 1) How do environmental cues from a given interface affect users' spatial memory and broader situational awareness? 2) How does spatial memory differ across individuals, and can those differences be attenuated via affordances within a synthetic environment? The virtual camera and texture-based approach described above offers researchers a flexible platform for exploring these research questions.

ACKNOWLEDGMENTS

This work was supported by a grant from the U.S. Air Force Office of Scientific Research as well as the U.S. Army Research Lab.

REFERENCES

[1] Anguelov, D., Dulong, C., Filip, D. *et al.*, "Google Street View: Capturing the World at Street Level," Computer, 43(6), 32-38 (2010).
[2] Vincent, L., "Taking Online Maps Down to Street Level," Computer, 40(12), 118-120 (2007).
[3] Szeliski, R., "Video mosaics for virtual environments," Computer Graphics and Applications, IEEE, 16(2), 22-30 (1996).
[4] Greenhill, S., and Venkatesh, S., "Virtual observers in a mobile surveillance system," Proceedings of the 14th annual ACM international conference on Multimedia, 579-588 (2006).
[5] Kadous, M. W., Sheh, R. M., and Sammut, C., "Effective user interface design for rescue robotics," 1st ACM Conference on Human-Robot Interaction, 250-257 (2006).
[6] Meguro, J., Hashizume, T., Takiguchi, J. *et al.*, "Development of an autonomous mobile surveillance system using a network-based RTK-GPS," IEEE International Conference on Robotics and Automation, 3096-3101 (2005).
[7] Boonsuk, W., Gilbert, S., and Kelly, J. W., "The Impact of Three Interfaces for 360-Degree Video on Spatial Cognition," Proceedings of the 2012 Annual Conference on Human Factors in Computing Systems, (2012).
[8] Burt, P., and Adelson, E., "A multiresolution spline with application to image mosaics," ACM Transactions on Graphics, 2(4), 217-236 (1983).
[9] Foote, J., and Kimber, D., "FlyCam: Practical panoramic video and automatic camera control," IEEE International Conference on Multimedia and Expo, 3, 1419-1422 (2000).
[10] Sun, X., Foote, J., Kimber, D. *et al.*, "Panoramic video capturing and compressed domain virtual camera control," Proceedings of the ninth ACM international conference on Multimedia, 329-347 (2001).
[11] Shum, H. Y., and Szeliski, R., "Panoramic image mosaics," Microsoft Research Technical Report, (1997).
[12] Szeliski, R., "Image mosaicing for tele-reality applications," DEC and Cambridge Research Lab Technical Report, (1994).
[13] Park, J., and Myungseok, A., "A novel application of panoramic surveillance system," IEEE International Syposium on Industrial Electronics, 205-210 (2009).
[14] Tang, W.-K., Wong, T.-T., and Heng, P. A., "A system for real-time panorama generation and display in tele-immersive applications," Multimedia, IEEE Transactions on, 7(2), 280-292 (2005).
[15] Micusik, B., and Kosecka, J., "Piecewise planar city 3D modeling from street view panoramic sequences," Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, 2906-2912 (2009).
[16] Czerwinski, M., Tan, D. S., and Robertson, G. G., "Women take a wider view," Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves, 195-202 (2002).
[17] Shaurette, M., [Wireless Webcam Field Trips - Construction Students Tour Jobsites Without Leaving the Classroom] VDM Verlag, Saarbrücken, Germany, (2009).
[18] Kaibel, A., Auwärter, A., and Kravčík, M., "Guided and Interactive Factory Tours for Schools: Innovative Approaches for Learning and Knowledge Sharing," in W. Nejdl and K. Tochtermann (Eds.), Springer Berlin / Heidelberg, 198-212 (2006).
[19] Barshinger, T., and Ray, A., "From Volcanoes to Virtual Tours: Bringing Museums to Students through Videoconferencing Technology," Annual Conference on Distance Teaching & Learning, (1998).
[20] Banitsas, K. A., Georgiadis, P., Tachakra, S. *et al.*, "Using handheld devices for real-time wireless teleconsultation," Engineering in Medicine and Biology Society, 2004. IEMBS '04. 26th Annual International

Conference of the IEEE, 2, 3105-3108 (2004).
[21] Bauer, M., Heiber, T., Kortuem, G. *et al.*, "A collaborative wearable system with remote sensing," Wearable Computers, 1998. Digest of Papers. Second International Symposium on, 10-17 (1998).
[22] Kelly, J., and McNamara, T., "Spatial memories of virtual environments: How egocentric experience, intrinsic structure, and extrinsic structure interact," Psychonomic Bulletin & Review, 15(2), 322-327 (2008).
[23] Loomis, J. M., and Knapp, J. M., "Visual perception of egocentric distance in real and virtual environments," in L. J. Hettinger and M. W. Haas (Eds.) [Virtual and Adaptive Environments], Elrbaum, Mahwah, NJ, 21-46 (2003).
[24] Richardson, A., Montello, D., and Hegarty, M., "Spatial knowledge acquisition from maps and from navigation in real and virtual environments," Memory & Cognition, 27(4), 741-750 (1999).
[25] Williams, B., Narasimham, G., Westerman, C. *et al.*, "Functional similarities in spatial representations between real and virtual environments," ACM Trans. Appl. Percept., 4(2), 12 (2007).