First Person Indoor/Outdoor Augmented Reality Application: ARQuake

Bruce Thomas, Ben Close, John Donoghue, John Squires, Phillip De Bondi and Wayne Piekarski

Wearable Computer Laboratory, School of Computer and Information Science, University of South Australia, Mawson Lakes, South Australia

Abstract: This paper presents a first person outdoor/indoor augmented reality application ARQuake that we have developed. ARQuake is an extension of the desktop game Quake, and as such we are investigating how to convert a desktop first person application into an outdoor/indoor mobile augmented reality application. We present an architecture for a low cost, moderately accurate six degrees of freedom tracking system based on GPS, digital compass, and fiducial vision-based tracking. Usability issues such as monster selection, colour, input devices, and multi-person collaboration are discussed.

Keywords: Augmented reality; Computer games; Wearable computers

1. Introduction

Many current applications place the user in a first-person perspective view of a virtual world [1], such as games, architectural design viewers [2], geographic information systems and medical applications [3,4]. In this paper, we describe a project to move these forms of applications outdoors, displaying their relevant information by augmenting reality. In particular we consider the game Quake (idSoftware). As with other researchers [5], we wish to place these applications in a spatial context with the physical world, which we achieve by employing our wearable computer system Tinmith [6–8]. Tinmith is a context-aware wearable computer system, allowing applications to sense the position of the user's body and the orientation of the user's head. The technique we are developing will genuinely take computers out of the laboratory and into the field, with geographically-aware applications designed to interact with users *in the physical world*, not just in the confines of the computer's artificial reality. The key to this exciting practical technology is *augmented reality* (AR). Users view overlaid computer-generated information by means of see-through head-mounted displays. Unlike virtual reality, where the computer generates the entire user environment, augmented reality places the computer in a relatively unobtrusive, assistive role.

In the ARQuake application, a simplified representation of the physical world gaming location is modelled as a Quake 3D graphical model. The augmented reality information (monsters, weapons, objects of interest) is displayed in spatial context with the physical world. The Quake model of the physical world (walls, ceiling, floors) is not shown to the user: the see-through display allows the user to see the actual walls, ceilings and floors which ARQuake need only model internally. Coincidence of the actual structures and virtual structures is key to the investigation; the AR application models the existing physical outdoor structures, and so omission of their rendered image from the display becomes in effect one of our rendering techniques.

1.1. Aims

Our *aim* is to construct first-person perspective applications with the following attributes:

1. The applications are situated in the physical world.
2. The point of view that the application shows to the user is completely determined by the position and orientation of the user's head.

© Springer-Verlag London Ltd
Personal and Ubiquitous Computing (2002) 6:75–86

Mullen Industries, LLC - Ex. 1021
Page 1

3. Relevant information is displayed as augmented reality via a head-mounted see-through display.
4. The user is mobile and able to walk through the information spaces.
5. The applications are operational in both outdoor and indoor environments.
6. The user interface additionally requires only a simple hand-held button device.

1.2. Research issues

To achieve these aims, we investigated a number of research issues in the areas of user interfaces, tracking, and conversion of existing desktop applications to AR environments.

User interfaces for augmented reality applications that simultaneously display both the physical world and computer-generated images require special care. The choice of screen colours for the purely virtual images that the application must display requires attention to the lighting conditions and background colours of the outdoors. The keyboard and mouse interactions must be replaced with head/body movement and simple buttons. The screen layout of the user interface must accommodate the AR nature of the application.

The six degrees of freedom (6DOF) tracking requirements for these forms of applications must be addressed. We require a low cost, moderately accurate 6DOF tracking system. Tracking is required for indoor and outdoor environments over large areas, for example our usual testing environment is our campus [9]. GPS positional error has a less noticeable effect for the registration of augmented reality information at distance, but we need to address positional error when registering augmented information at close distances (< 50 m). Such a tracking system could be used for other applications, such as tourism information, visualisation of GIS information, and architectural visualisation.

It is also necessary to modify the Quake game to accommodate the AR nature of the new application. The user's movement changes from a keystroke-based relative movement mode to a tracking-based absolute mode. The game's coordinate system must be calibrated to the physical world. Finally, the field of view of the display must be calibrated to the physical world.

2. Background

There are two basic styles of tracking: absolute and relative. Furthermore, machine learning can train a system to recognise locations in a building or outdoors. Golding and Lesh use an array of sensors: accelerometers, magnetometers, temperature and light to track a user in a set of known *locations* [10]. Aoki, Schiele and Pentland [11] use a camera to train the system to recognize the user's location and approaching trajectory. These systems can determine its present room, and whether it is entering or leaving that room.

Previous research has established that outdoor tracking with inexpensive differential GPS and commercial grade magnetic compasses are inaccurate for augmented reality applications [12]. Traditional hybrid approaches combine a number of different systems such as inertial, optical, electro-magnetic and GPS. In this paper, we present our hybrid approach of combining vision-based optical tracking with GPS and a magnetic compass.

A number of researchers are investigating *fiducial* vision-based tracking [3,13]. We based our optical tracking system on the fiducial marker tracking system ARToolKit developed by Kato and Billinghurst [14]. The ARToolKit is a set of computer vision tracking libraries that can be used to calculate camera position and orientation relative to physical markers in real

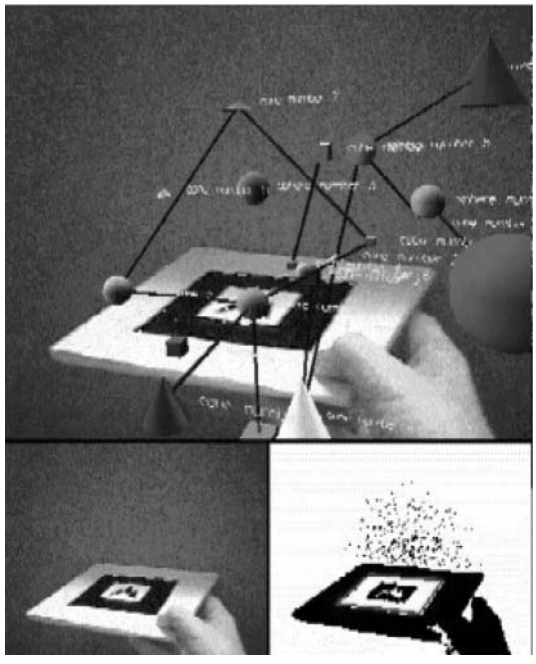

Fig. 1 Example of a fiducial marker.

Mullen Industries, LLC - Ex. 1021
Page 2

time. ARToolKit features include the use of a single camera for position/orientation tracking, fiducial tracking from simple black squares, pattern matching software that allows any marker patterns to be used, calibration code for video and optical see-through applications, and sufficiently fast performance for real-time augmented reality applications.

The fiducial markers are known-sized squares with high contrast patterns in their centres. Figure 1 shows an example marker. The ARToolKit determines the relative distances and orientation of the marker from the camera. In addition, the ARToolKit incorporates a calibration application to determine the placement of the camera relative to the user's line of sight; thus the ARToolKit can determine proper placement of graphical objects for AR applications.

2.1. The original Quake game

We chose Quake as the primary application for a number of reasons. Quake fits the general model of AR that we are studying, as it is a first-person 3D application with autonomous agents to interact with the user. We were able to obtain the application source code. Finally, the Quake graphics engine is very quick and runs on a wide range of computing platforms and operating systems.

Quake is a first-person *shoot 'em up* game. Quake has two stated goals: "First, stay alive. Second, get out of the place you're in" (idSoftware). The user interface is based around a single, first-person perspective screen. The large top part of the screen is the view area, showing monsters. Status information is immediately below at the bottom of the screen.

One moves around Quake in one of four modes: walking, running, jumping or swimming, and performs one of three actions: shooting a weapon, using an object, or picking up an object. Weapons are aimed by changing the view direction of the user, and fired by pressing a key. To push a button or open a door, the user walks up to the door or button. A user picks up items by walking over them. Part of the challenge of the game is finding special objects like buttons, floor-plate doors, secret doors, platforms, pressure plates and motion detectors. Quake incorporates platforms that move up and down, or follow tracks around rooms or levels. Pressure plates and motion detectors may be invisible or visible, and there are sensors which open doors, unleash traps, or warn monsters.

Fig. 2. Wearable computer platform.

2.2 Wearable computer platform

The Tinmith wearable computer system hardware is all mounted on a rigid backpack so that the items can be attached firmly, (see Fig. 2). Processing is performed by a Toshiba 320CDS notebook (Pentium-233, 64 Mb RAM) running the freely available LinuxOS and associated programs and development tools. The laptop is very generic, and not even the latest in available CPUs, so another computing unit could be substituted. The limited I/O capabilities of the single serial port are augmented with the use of a four serial port Quatech QSP-100 communications card. Connected to the laptop are a Precision Navigation TCM2-80 digital compass for orientation information (we now use an Intersense 300 tracker for head orientation), a Garmin 12XL GPS receiver for positioning, and a DGPS receiver for improved accuracy. For the Head Mounted Display (HMD), we use alternately the i-Glasses unit from I-O Display Systems, and the Sony Glasstron PLM-S700E. Various other devices are present as well, such as power converters for the different components,

Mullen Industries, LLC - Ex. 1021
Page 3

necessary connection cabling, and adaptors. The construction of the backpack was directed with ease of modifications in mind, at the sacrifice of wearability and miniaturisation.

The Tinmith system [8] supports outdoor augmented reality research. The system is comprised of a number of interacting libraries and modules. A number of software libraries form a support base for writing code in the system: a graphics interface on top of X windows; an interface to coordinate/datum transformations and numeric conversions; encode/decode libraries for transmitting structures over a network; tools for network communications and high level I/O; low level interfaces to Unix system calls, asynchronous I/O code, string handling, event generation, and error checking.

3. Using ARQuake

The goal of ARQuake was to bring the intuitive nature of VR/AR interfaces into an indoor/outdoor game. A user first dons the wearable computer on their back, places the HMD on their head, and holds a simple two-button input device. The user then performs a simple calibration exercise to align the HMD with their eyes, and then they start playing the game. All of the keyboard and mouse controls have been replaced with position/orientation information and a two-button haptic gun input device. As movement aspects of the game have been engineered to fit the physical world, there is no concept of commands to walk, run, jump, swim, or of moving platforms. The user's own movement determines the rate and direction of movement. The remainder of this section describes the Quake level we developed and its user interaction.

3.1. Haptic gun

To improve the play-ability of ARQuake, we replaced mouse and keyboard button presses with a haptic gun device. The aiming of the weapons is still the direction of the user's head, but the firing of the weapon and changing of weapons is performed with button presses on the new gun input device. To give the gun a "recoil" feel, we installed simpe haptic feedback into the gun.

A haptic gun was developed from an appropriate toy plastic gun (see Fig. 3). Two standard commercial push buttons were installed, along with a micro-switch to replace the primitive trigger switch. A solenoid was placed towards the rear of the gun, behind the centre of gravity, to enhance the "pitch-up" sensation of the recoil. A vibrating motor was placed as far forward in the gun as possible to increase the moment arm from the centre of gravity, thereby enhancing the effect. The sound effects are generated by the solenoid and vibrating motor.

Fig. 3. Haptic gun.

The gun provides a number of haptic sensations and sounds. There is the single shot, which provides a strong recoil with a loud bang. The number of single shot weapons allow for additional shots to be fired after a suitable reload time. The amount of time for reloading varies between single shot weapons. The shotgun provides a double shot haptic and sound effect; this effect simulates the rapid firing of both barrels serially. The multiple firing weapons, such as the machinegun, provide a less strong recoil with a short time interval between firing. The sound effect is a higher pitch bang with a lower volume. Finally, there is an energy weapon that fires a continuous stream of energy; the haptic effect is a continuous vibration of the gun with a high pitch whining sound effect.

Mullen Industries, LLC - Ex. 1021
Page 4

3.2. Monsters

There are 16 different types of monster in the Quake world. Some have attributes that make them unsuitable for inclusion in our AR version of the game. Because of the limitations on movement imposed by the tracking hardware, the best monsters were those that walked or leaped and those that were relatively easy to destroy and did not inflict extreme damage on the user with their first attack.

We chose seven types of monsters to be included in our game world. These monsters types are all land-based creatures that use weapons from a distance. The monsters' *skin* colour and texture were changed to make them easier to see and distinguish from the physical world. The choice of colours used in the texture maps or skins of the monsters are based on the user testing described later.

We excluded monsters which were too large for the environment or which had unexpected effects; those which swam, as our campus does not include water features; those which flew, they move too quickly; those which surround the user or have some other interaction which would require haptic feedback; and those whose attack tended to be immediately fatal, as they are not enjoyable.

3.3. Campus level

We created a Quake level (game world) representing the Mawson Lakes campus of the University of South Australia. The walls in Quake are the external walls of the campus buildings and the interior walls of the Wearable Computer Laboratory (WCL). The walls are rendered in two fashions, black for game mode and a grid patterned for testing mode. In both these modes, the walls occlude the graphic objects in Quake that may be located behind the walls. As described earlier, in the game mode black walls are transparent to the users during the game. The Quake graphics engine renders

Fig. 4. Quake campus level.

Mullen Industries, LLC - Ex. 1021
Page 5

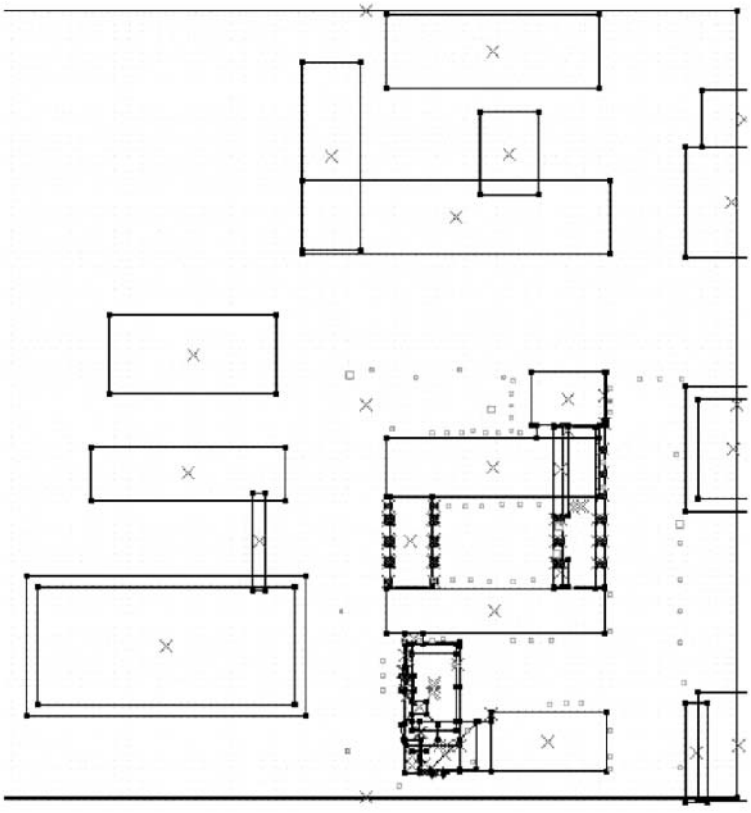

Fig. 5. One quarter of the Quake campus level.

only monsters, items on the ground, and regions of interest. This Quake level was derived from architectural drawings of the campus provided by the university; where the architect's drawings had become incorrect, we surveyed those portions ourselves.

The size of the outside modelled area is 500 metres (East/West) by 500 metres (North/South). Figure 4 depicts a top-down view of the level we created, and Fig. 5 is a detailed view of the most interesting quarter of the map. We have placed over 200 monsters in the level as follows: on the ground, on top of buildings, and in second floor windows. There are hundreds of items placed on the ground for the user to pick up: pieces of armour, rockets, rocket launchers, shotgun shells and health boxes.

The system of tracking used in this system tends to make the user less agile than the "super-human" agility found in the normal game. Therefore we have included more support equipment than would be found in the normal game: armour, weapons and ammunition.

3.4. Walking around

Once the system is up and running, the user moves through the level by walking, and changes view by looking around. The user views the game and the physical world through the HMD, an example is shown in Fig. 6 and Fig. 7. The bottom portion of the screen is a status bar containing information about armour, health, ammo and weapon type. The majority of the screen is reserved for the AR images of monsters and game objects (see Fig. 8). In the original Quake, certain actions are performed by the user being in a certain proximity to a location in a Quake level. We have retained most of those actions. Users pick up objects as in the original Quake by walking over them. Traps are triggered by standing in or moving through predetermined locations. Actions that are not easily reflected in the physical world are removed from the game, such as secret and locked doors.

The tracking of the user's position and orientation of the user's head handles the majority of the interaction for the user. The

Mullen Industries, LLC - Ex. 1021
Page 6

Fig. 6. User's heads up display.

Fig. 7. Second user's heads up display.

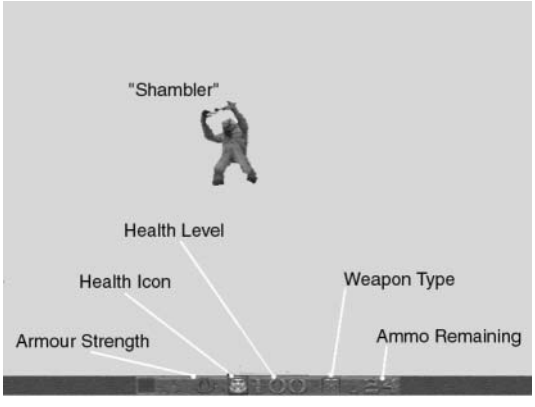

Fig. 8. Status bar.

only other interactions for the user to perform are to shoot or change the current weapon. We employ a two-button (thumb button and index finger button) hand-held device as a physical input device for these actions. The thumb button is used to change weapons, and the index finger button fires the current weapon. The direction the weapon fires is the centre of the current view of the HMD.

Through informal user's studies, users thought that the visibility of the ARQuake system was good; however many of the users found that bright sunlight made seeing through the display difficult. Despite only using the system once, the users found the hand held input device intuitive, easy to use, and very quick to learn. A few of the users found themselves pointing the device in a gun like fashion when firing at the targets. No one reported feeling nauseated while using the system. Users believed that it was easy to pick up items although it was difficult to tell when an item had been picked up without some form of confirmation.

People disliked the colours on the status bar and thought the range of colours were limited. The monster colours were good and easy to see, and the users were able to easily identify monsters. When asked "Is the movement in the augmented reality relative to the real world?" Most people thought that the movement relative to the real world was okay but commented on the lag and jitter when rotating their heads. When asked "Is it easy to shoot at the monsters?" Most subjects found that the lag made it difficult to align the cross hairs at the targets. The actual process of firing the weapon was easy.

3.5. Field of view

Even if the alignment of the Quake world with the physical world is exact, an incorrect perspective or field of view will be highlighted as inconsistencies in the virtual world. The default field of view for the game is 90 degrees (45 degrees each side), allowing a reasonable coverage of the world to fit onto a computer screen. This field of view unfortunately suffers from the *fish eye* distortion effect when comparing the objects in the Quake world with real objects. The HMD we are using, I-Glasses, has approximately a 25-degree horizontal field of view. The only calibration adjustment for the HMD with Quake is changing the game's field of view setting and scaling of the graphical objects. We are currently using a field of view value of 25 degrees, but there are artifacts introduced as in the user is positioned farther forward. We are investigating the graphics model of Quake to determine how it differs from traditional graphics models.

Mullen Industries, LLC - Ex. 1021
Page 7

4. Tracking

As previously stated, one of the goals of the system is to provide continuous indoor and outdoor tracking. The system tracks through the combination of a GPS/compass system and with a vision-based system. Our tracking needs are categorized into three areas as follows: outdoors far from buildings, outdoors near buildings, and indoors. Each of these require a different approach, while maintaining position and orientation information in a common format of WGS 84/UTM positioning information and heading/pitch/roll angles for orientation information. The use of visual landmarks can improve registration in one of two ways, first to allow the system to correct the final image by aligning the landmark with a known position in the graphical image, and secondly to use the landmarks to extract a relative position and orientation of the camera from the landmarks. We have chosen the second option to investigate as it provides the most general tracking solution.

4.1. Outdoors away from buildings

GPS positional inaccuracies are less of a problem for our Quake application when a user is at a large distance >50 m from an object which requires registration, while orientation errors remain constant as to angular deviations in the user's field of view. An extreme example of how positional errors have a reduced registration error effect at distance is the using of the ARQuake game on a flat open field, where the system does not require graphics to be registered to any physical object except the ground. In this scenario there are no walls to occlude the monsters and items of interest. Since the game is slaved to the screen, what the user sees on the display is what the game believes is the user's current view. Therefore, the user's actions will perform correctly in the context of the game.

In the case where a building is visible but the user is a large distance from the building, the inaccuracies are low and therefore not distracting. The problems occur when the physical buildings do not occlude the game graphics properly. The visual effect of poor occlusion is that monsters appear to walk through walls or pop out of thin air, but at distance these errors do not detract from the game. Such occlusion problems exist but they are visually very minor, because the user is generally moving their head during the operation of the game. At 50 m a difference of 2–5 m (GPS tracking error) of the user's position is approximately a 2–5 degree error in user's horizontal field of view, and the compass itself has an error of ± 1 degrees.

4.2. Outdoors near buildings

When using ARQuake with the GPS/compass tracking less then 50 m from a building, the poor occlusion of monsters and objects near the physical buildings, due to GPS error, becomes more apparent. As the user moves closer to buildings, inaccuracies in GPS positional information become prevalent. The system is now required to slave the Quake world to the real world, and furthermore, in real time. As an example, when a user is ten metres from a building and their position is out by 2–5 m, this equates to an error of 11–27 degrees; this is approximately a half to the full size of the horizontal field of view of the HMD. When the error is greater than the horizontal field of view,the virtual object is not visible on the HMD.

Our proposed design is to enhance the accuracy when the user is near buildings using an extended version of ARToolKit. By using fiducial markers specifically engineered for outdoor clarity (approximately 1 m in size), and with each marker set up on a real world object with known coordinates, accurate location information can be obtained. Figure 9 shows what a fiducial marker on the corner of a building would look like in our Quake world. These markers provide a correction in the alignment of the two worlds. We are investigating the use of multiple fiducial markers to reduce

Fig. 9. Fiducial marker on a building.

Mullen Industries, LLC - Ex. 1021
Page 8

uncertainty due to marker mis-detection caused by lighting issues. Since the extended ARToolKit we are developing supplies positioning and orientation information in the same format as the GPS/compass system, ARQuake can transparently use either the GPS/compass or vision-based tracking systems. Our initial approach for determining when to use the information from the GPS/compass or the ARToolKit methods is use the ARToolKit's information first, when it is confident of registering a fiducial marker. As ARToolKit recognises a fiducial marker, the toolkit returns a confidence value, and the system will have a threshold of when to switch over to use the toolkit. When the confidence value goes below the threshold, the GPS/compass information is used.

4.3. Indoors

Our next proposed design caters for a user walking into a building with fiducial markers on the inside walls and/or ceilings; the tracking system starts using the vision-based component of the tracking system. This form of tracking is similar to the work of (Ward et al.) [15]. Our system is lower-cost and is not as accurate, but does keep tracking errors within the accuracy which our application needs, 2–5 degree of error in user's horizontal field of view. We are experimenting with placing markers on the walls and/or the ceilings.

When the markers are placed on the wall, we point the vision-based tracking camera forwards. It was necessary to size and position the patterns on the walls so that they would be usable by the system regardless of whether the user was very close or very far from the wall. In this case, we chose to use patterns that were a size of 19cm^2. From testing, we found that the system could register a pattern at a range of 22.5 cm to 385 cm from the wall. In a 8 by 7 m room this range would be sufficient (for the initial stages of the project) and an accuracy of within 10 cm at the longer distances. It is important that no matter where the user looks in the room that, at least one pattern must be visible to provide tracking. For this reason, we realised that to implement the patterns on the walls, as the sole means of tracking would require different size targets. We are investigating the use of targets themselves as patterns inside larger targets; therefore one large target may contain four smaller targets when a user is close to a target.

Fig. 10. Fiducial marker on the wall.

Our second approach has been to place the markers on the ceiling, with the vision-based tracking camera pointed upwards. The camera does not have the problem of variable area of visible wall space, as the distance to the ceiling is relatively constant. The main differences are the varying heights of users. In the first instance we are implementing a small range of head tilt and head roll (± 45 degrees). Perspectives such as those from laying down or crawling will be investigated in the future.

The patterns on the ceiling were placed so that at one time the tracking software could reliably identify at least one pattern. With the camera mounted on the backpack at height of 170 cm and with the room height of 270 cm, our current lens for the camera views a boundary of at least 130 cm^2; we chose a pattern of 10 cm^2 in size.

4.4. Choosing colours

The choice of colours is important for outdoor augmented reality applications, as some colours are difficult to distinguish from natural surroundings or in bright sunlight. The original Quake game incorporates a "dark and gloomy" colour scheme to give the game a foreboding feelings. Dark colours appear translucent with the see-through HMDs. Monsters and items need different colours to be more visible in an outdoor environment. We ran a small informal experiment to determine a starting point in picking colours for use in an outdoor setting. This informal study was to gauge the visibility and opaqueness of solid filled polygons display of the

Mullen Industries, LLC - Ex. 1021
Page 9

see-through HMD. We are interested in which colours to texture large areas of the monsters and items in the game. These colours are not necessarily appropriate for textual or wire-frame information. Further studies are required for these and other forms of AR information.

The testing method was to view different colours in one of four conditions: (1) standing in shade and looking into the shady area; (2) standing in shade and looking into a sunny area; (3) standing in a sunny area and looking at a shady area; and (4) standing in a sunny area and looking to a sunny area. We tested 36 different colour and intensity combinations, nine different colours (green, yellow, red, blue, purple, pink, magenta, orange and cyan) and four different intensities. The testing was performed outside with the Tinmith wearable computer using the I-Glasses see-through HMD. The colour/intensity combinations were scored for visibility and opaqueness in each of the four viewing conditions on a scale of one (very poor) to ten (very good)).

Our strongest set of criteria for colour/intensities were both a mean score of at least seven over the four viewing conditions, as well as a minimum score of six on each of the conditions. Nine colours satisfy this quality level: three shades of purple, two shades of blue, two shades of yellow, and two of green.

A more in depth look at when standing in shade and looking into the shade shows the best colours were determined to be bright purple and bright magenta. When standing in shade but looking into a sunny area, the best colours were determined to be a dark green and pink. The colour to avoid is dark orange, which was given a score of 4. All other colours were a score of okay (5) and above. The results changed when the subject was moved out into a sunny area. When standing in a sunny area and looking to a shady area, bright yellow and bright purple scored the best. There were eight colours that scored a poor; these are as follows: bright yellow, bright red, bright pink, bright magenta, bright orange, bright cyan, and dark cyan. Finally, the results of the best colours for when the subject was standing in the sun and looking into a sunny area were as follows: bright green, three shades of blue and dark purple. There were 17 colours that score a poor (3 or 4). The colours to avoid are all intensities of cyan, orange, magenta, pink, and red.

5. Collaboration

The original Quake engine allows multiple people to play the game simultaneously, and allows communication between players via a text-based mechanism. To avoid a user having to divert attention from what is occurring in the world around them, we provide the users a facility for two-way voice communication. We have add a number of collaboration features to the ARQuake game.

5.1. Multi-player

To allow both users on a desktop computer and users on a wearable computer to use the system simultaneously, two different representations of the maps are required. For example the users with the wearable will not require the buildings to be rendered, as they will actually see the physical buildings in the augmented reality. Alternatively the users who are indoors will require the buildings to be textured, as they will not be able to see the physical buildings from their stationary position. Both representations of the maps require the buildings' placement, size and orientation to be identical on both computer platforms.

As previously mentioned, multi-player from a desktop platform to a second desktop platform is already implemented in the original Quake software. We then extended this to a desktop platform to wearable platform multi-player configuration. We used a WaveLAN to network the desktop platform with the wearable platform. At present we have been able to have two players using the WaveLAN, one on the wearable and one on a desktop. The game was able to run smoothly using the WaveLAN network.

5.2. Using a Pointing device to aid communication

In ARQuake there is no method for a person to point at a desired object. The avatar, which represents them, is not capable of displaying a pointing action to other users who may be in the system. This is not an issue if all players are using wearable computers and are using the AR system. The players can see other players physically pointing. However in the case where multiple users are on a desktop platform and a wearable platform, a method of presenting an artificial method of pointing is required. The

Mullen Industries, LLC - Ex. 1021
Page 10

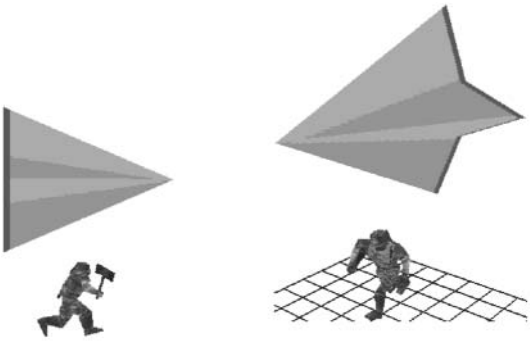

Fig. 11. 3D pointing device.

system should allow a user to point out objects and with speech we believe that a user will be able to accurately indicate an object. The use of speech and social protocols will allow the confirmation of the correct direction or object being referring to. The pointer is able to guide other users rather than specify an exact object.

To make this pointing device useable, it needs to be easily seen from all angles. We designed the pointer as a 3D pyramid with dented walls (see Fig. 11). The pointer allows other users looking at the shape the ability to gauge where the other user is referring. Initially the shape was one colour, but we found that in some cases it was difficult to tell which direction the arrow was pointing. We then modified the pointing device to have one end textured a different colour, to allow the user to establish which direction the pointer is facing at all angles.

A future development is to allow the user to control the visibility and independent movement of the pointer. The pointer currently mimics the behaviour of the player virtual avatar, and allows a full range of pointing on both the x and y planes.

6. Implementing ARQuake

The original Tinmith software did have not required any modifications to support the extensions mentioned in this paper, but two additional modules `modpatt` and `modquake` have been added to provide the new features.

The `modpatt` module performs the pattern recognition using the ARToolKit, and also reads in position and orientation values. (More details concerning architecture and implementation of the existing Tinmith modules can be found in Piekarski et al. [6,8]. The modpatt module uses the pattern recognition extended from ARToolKit to refine the position and orientation information, which is then passed back to the Tinmith system for the other modules to process. In the case of when the system is indoors, `modpatt` is responsible for using just the camera recognition to generate position information, as the GPS does not function indoors.

The `modquake` module extracts information from the Tinmith system, such as position and orientation, and converts this into UDP packets, which are then sent onto the local network. The modified Quake program then receives these UDP packets, and converts the data into Quake's local coordinate system, by scaling and translating. Quake then uses this information to render the display at the appropriate location with the correct head position, and the player can control the game using physical movement.

7. Conclusion

This paper reports on an ongoing research project – ARQuake, an outdoor/indoor augmented reality first-person application. Although the implementation has not been completed, many interesting results have been found, in particular user interface issues for AR outdoor/indoor augmented reality games; an architecture for low cost, moderately accurate indoor/outdoor 6DOF tracking; and implementation issues for converting a desktop application into an AR application.

The current ARQuake game is running on our wearable computer platform with the 6DOF GPS/compass tracking system. The interaction of the game was operating within the accuracy of this tracking system. We have modelled an outdoor section of our campus and interior of the WCL. The graphics of the game are running at approximately 30 frames per second, with GPS updates once per second and compass updates at 15 times per second.

The results of our research to this point in the implementation, include the design and implementation of a user interface for an augmented reality application, and proposed some guidelines. Specifically, identified by informal experimental evidence, a set of colours that take into account different lighting and background colours for outdoors use. We successfully mapped the keyboard and mouse interactions of the ARQuake application to head/body movements

Mullen Industries, LLC - Ex. 1021
Page 11

and a simple two-button input device. A simple user interface layout was incorporated into the ARQuake application. The game is quite playable in this configuration, and we are continuing to investigate how to overcome difficulties with registration errors when indoors and close to buildings.

We proposed and described a low cost moderately accurate indoor/outdoor 6DOF tracking system, by a novel combination of optical tracking and GPS/compass tracking to provide an absolute tracking system.

The solution is not restricted to overcoming registration problems, but it does provide a general tracking solution. Fiducial markers are placed at the corners of buildings to provide more accurate positioning information for ARQuake. When a fiducial marker comes into view of a head mounted camera, this more accurate tracking information is used instead of the GPS/compass data. The vision based tracking is not fully functional, and we are continuing to investigate and develop this solution.

Aside from the specific technical achievements of our work to date, it is perhaps most important to point out that our work can serve as proof that augmented reality is readily achievable with inexpensive, off-the-shelf software. The translation of the application set from the desktop to incorporate the physical world brings closer the possibility of truly ubiquitous computing.

Acknowledgments

For their support of this project, we would like to thank the Advanced Computing Research Centre of the University of South Australia; John Maraist; Arron Piekarski; and the Defence Science and Technology Organisation.

References

1. Klinker GJ, Ahlers KH et al., Confluence of computer vision and interactive graphics for augmented reality. PRESENCE: Teleoperations and Virtual Environments 1997; 6(4): 433–451
2. Brooks FP, Walkthrough – a dynamic graphics system for simulating virtual buildings. Workshop on Interactive 3D Graphics, 1986
3. State A, Hirota G et al., Superior augmented reality registration by integrating landmark tracking and magnetic tracking. Proceedings of SIGGRAPH 1996, New Orleans, LA, ACM, 1996; 439–446
4. State A, Livingston MA et al., Technologies for augmented-reality systems: realizing ultrasound-guided needle biopsies. Proceedings of SIGGRAPH 1996, New Orleans, LA. ACM, 1996.
5. Hollerer T, Feiner S et al., Situated documentaries: embbeding multimedia presentations in the real world. 3rd International Symposium on Wearable Computers, San Francisco, CA, 1999; 79–86
6. Piekarski W, Gunther B et al., Integrating virtual and augmented realities in an outdoor application. Proceedings of the Second IEEE and ACM International Workshop on Augmented Reality (IWAR) '99, San Francisco, CA, 1999
7. Piekarski W, Thomas BH, Tinmith-Metro: new outdoor techniques for creating city models with an augmented reality wearable computer. Fifth International Symposium on Wearable Computers, Zurich. IEEE, 2001
8. Piekarski W, Thomas BH et al., An architecture for outdoor wearable computers to support augmented reality and multimedia applications. Proceedings of the Third International Conference on Knowledge-Based Intelligent Information Engineering Systems, Adelaide, IEEE,1999
9. Neumann U, You S et al., Augmented reality tracking in natural environments. International Symposium on Mixed Realities, Tokyo, Japan, 1999
10. Golding AR, Lesh N, Indoor navigation using a diverse set of cheap, wearable sensors. 3rd International Symposium on Wearable Computers. San Francisco, CA, 1999; 29–36
11. Aoki H, Schiele B et al. Realtime positioning system for a wearable computers. 3rd International Symposium on Wearable Computers, San Francisco, CA, 1999
12. Azuma RT, The challenge of making augmented reality work outdoors. First International Symposium on Mixed Reality (ISMR '99), Yokohama, Japan. Springer-Verlag, 1999
13. Park J, Jiang B et al., Vision-based pose computation: robust and accurate augmented reality Tracking. Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality '99, San Francisco, CA, 1999; 3–12
14. Kato H, Billinghurst M, Marker tracking and HMD calibration for a video-based augmented reality conferencing system. Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality '99, San Francisco, CA, 1999; 85–94
15. Ward M, Azuma R et al., A demonstrated optical tracker with scalable work area for head-mounted display systems. Proceedings of 1992 Symposium on Interactive 3D Graphics. Cambridge, 1992

Correspondence to: Dr B. Thomas, Wearable Computer Laboratory, School of Computer and Information Science, University of South Australia, Mawson Lakes, SA 5095 Australia. Email: thomas@uniSA.edu.au

Mullen Industries, LLC - Ex. 1021
Page 12