Georeferenced Registration of Construction Graphics in Mobile Outdoor Augmented Reality

Amir H. Behzadan, A.M.ASCE[1]; and Vineet R. Kamat, M.ASCE[2]

Abstract: This paper describes research that investigated the application of the global positioning system and 3 degree-of-freedom (3-DOF) angular tracking to address the registration problem during interactive visualization of construction graphics in outdoor augmented reality (AR) environments. The global position and the three-dimensional (3D) orientation of a user's viewpoint are tracked, and this information is reconciled with the known global position and orientation of superimposed computer-aided design (CAD) objects. Based on this computation, the relative translation and axial rotations between the user's viewpoint and the CAD objects are continually calculated. The relative geometric transformations are then applied to the CAD objects inside a virtual viewing frustum that is coincided with the real world space that is in the user's view. The result is an augmented outdoor environment where superimposed graphical objects stay fixed to their real world locations as the user navigates. The algorithms are implemented in a software tool called UM-AR-GPS-ROVER that is capable of interactively placing static and dynamic 3D models at any location in outdoor augmented space. The concept and prototype are demonstrated with an example in which scheduled construction activities for the erection of a structural steel frame are graphically simulated in outdoor AR.

DOI: 10.1061/(ASCE)0887-3801(2007)21:4(247)

CE Database subject headings: Global positioning; Tracking; Transformation; Construction sites; Graphic methods.

Introduction

Augmented reality (AR) is the superimposition of computer-generated images over a user's view of the real world. By presenting contextual information in textual or graphical format, the user's view of the real world is enhanced or augmented beyond the normal experience. The addition of such contextual computer-generated information spatially located relative to the user can assist in the performance of several scientific and engineering tasks. For this reason, AR enabling technologies have been researched in an increasing number of studies in recent years (e.g., Azuma 1997; Lawson and Pretlove 1998; Azuma et al. 2001; Dodson et al. 2002; Piekarski and Thomas 2002).

Description of Problem

Graphical simulation is a powerful tool that can be used to effectively plan and design construction activities and operations. Using advanced graphical simulation tools, constructors can plan and study their project operations in detail and identify potential problems that may occur during construction (e.g., resource utilization, physical constraints, operational conflicts, etc.) before committing real resources in the field. For this reason, graphical simulation has been the focus of interest in many previous research studies in construction (e.g., Op den Bosch 1994; Dharwadkar et al. 1994; Kamat and Martinez 2005). Notwithstanding the research advances, a major practical challenge deters widespread adoption of graphical simulation in construction practice. That challenge is the time and effort required for model engineering.

The task of acquiring, cleaning, updating, and versioning computer-aided design (CAD) models of simulation objects for use in graphical simulation is called model engineering (Brooks 1999). In a graphical simulation of a construction operation, the required CAD models include the terrain (landscape), existing structures (roads, buildings, etc.), partially complete facility, equipment (trucks, cranes, etc.), materials (steel beams, concrete blocks, etc.), and humanoids (workers). CAD models of equipment, materials, and humanoids can be obtained from model vendors. Since they are generic, the models can be compiled into a library and repeatedly reused in simulations (Min et al. 2003). In addition, CAD models of facilities under construction can often be inherited from design (Brooks 1999).

The major model engineering problems in construction are evolving jobsite terrains and existing structures. In addition, CAD models inherited from design are often transmitted in a single or limited number of CAD layers (pieces), limiting their value in portraying a partially complete facility in graphical simulations of construction activities. Since each project is unique in location and design, the time and effort needed to create CAD models of the local environment (terrain, existing structures, and partially complete facility) is phenomenal. Model engineering is, by definition, very demanding and time consuming, and has been identified as one of the most significant deterrents to widespread

---

[1] Ph.D. Candidate, Dept. of Civil and Environmental Engineering, Univ. of Michigan, 2340 G.G. Brown, 2350 Hayward, Ann Arbor, MI 48109. E-mail: abehzada@umich.edu

[2] Assistant Professor, Dept. of Civil and Environmental Engineering, Univ. of Michigan, 2340 G.G. Brown, 2350 Hayward, Ann Arbor, MI 48109 (corresponding author). E-mail: vkamat@umich.edu

Note. Discussion open until December 1, 2007. Separate discussions must be submitted for individual papers. To extend the closing date by one month, a written request must be filed with the ASCE Managing Editor. The manuscript for this paper was submitted for review and possible publication on December 2, 2005; approved on January 5, 2007. This paper is part of the ***Journal of Computing in Civil Engineering***, Vol. 21, No. 4, July 1, 2007. ©ASCE, ISSN 0887-3801/2007/4-247–258/$25.00.

acceptance of graphical simulation in scientific problem solving in general (Brooks 1999), and in graphical simulation of construction activities in particular (Zhang et al. 2002).

The ability to graphically simulate construction activities in AR can be of significant help in alleviating the noted model engineering problems. In AR, a user views the real world, along with computer generated graphics superimposed or blended with that view (Barfield and Caudell 2001). Instead of completely replacing the real world as is done in virtual reality (VR), the idea in AR is to supplement it with relevant information, and to make the real and virtual objects coexist in augmented space.

The real advantage in AR is that the view of the real world (landscape, vehicles, people, etc.) can be used as a readymade backdrop for displaying superimposed information (e.g., graphics) of interest. An AR user only needs to create information (e.g., CAD models) that is augmented onto the real world view; s/he does not need to be concerned about creating the whole world itself. Thus, engineers who want to graphically simulate construction activities at a jobsite can use that site's real environment and partially completed facility as a background, and graphically simulate only the activities that are being planned at the location.

Main Research Contributions

The primary problem in AR is the need for accurate registration. Objects in the real world and superimposed virtual objects must be properly aligned with respect to each other, or the illusion that the two coexist in augmented space is compromised. In AR, registration can be achieved and maintained by monitoring trackers (sensors) mounted on a user as s/he navigates and performs tasks in augmented space. Without accurate, real-time knowledge of the position and orientation of the user's viewpoint, correct registration is not possible (Barfield and Caudell 2001). Traditional tethered tracking systems are based on magnetic, inertial, or optical sensors and computer vision systems, and rely on installations of large devices or dense arrays of sensors mounted in enclosed areas (Koller et al. 1997; Welch et al. 1999; Bachmann et al. 2001).

Such systems are appropriate for indoor applications and have hitherto been used in many AR studies in architecture and construction. Such registration techniques are however unsuitable for outdoor AR environments such as typical construction sites primarily due to the unprepared nature of the pertinent space. The constraints include: (1) the inability to install and maintain a tracking sensor infrastructure; (2) the relatively large range of possible user navigation; and (3) variable lighting conditions that preclude use of vision based tracking.

The presented research addresses the issue of outdoor registration by investigating the use of the global positioning system (GPS) and 3 degree-of-freedom (3-DOF) angular trackers to design and implement an outdoor AR registration technique that allows unconstrained user navigation in graphical simulations of construction activities. At each frame in a graphical AR simulation, the global position and three-dimensional (3D) orientation of the user's viewpoint (i.e., longitude, latitude, altitude, heading, pitch, and roll) are tracked, and this information is reconciled with the known global position and orientation of superimposed CAD objects. Based on this computation, the relative translation and axial rotations between the user's viewpoint and the CAD objects are calculated. These geometric transformations are then applied to the CAD objects inside an OpenGL viewing frustum that is coincided with the real world space that is in the user's view at that instant.

The result is an augmented outdoor environment where superimposed CAD objects stay fixed to their real world locations as the user moves about freely on a construction site. The algorithms are implemented in a proof-of-concept prototype called UM-AR-GPS-ROVER that is capable of interactively placing static and dynamic 3D CAD models at any desired location in outdoor augmented space. The concept and prototype are demonstrated with an example in which scheduled construction activities for the erection of a structural steel frame are graphically simulated in outdoor AR.

Current State of Knowledge

AR technology has been hitherto studied for several scientific and engineering applications. For example, integration of AR in CAD/CAM systems helps manufacturing companies (e.g., automotive, airlines, etc.) to model mechanical designs, visualize stresses or flows calculated from previous simulations, test for interferences through digital preassembly, and study the manufacturability and maintainability of subsystems (Barfield and Caudell 2001).

Visualization of medical information projected onto a patient's body on the other hand, is also one of the established applications of AR technology. Traditional magnetic resonance imaging (MRI) and computed tomography (CT) images provide physicians with information on a totally detached display from the patient. Using AR displays allow MRI and CT images to be superimposed over the patient's anatomy which can assist in tasks such as the planning of surgical procedures (Barfield and Caudell 2001).

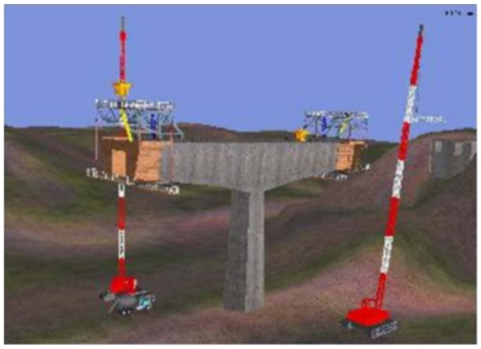
AR has also been used in military applications. For instance, the battlefield augmented reality system (BARS) is an AR system that can network multiple dismounted war fighters together with a command center. BARS also supports information gathering and human navigation for situation awareness in urban settings (Livingston et al. 2002).

Previous studies have also explored AR for many architecture and construction applications. For example, Webster et al. (1996) presented a system that shows locations of columns behind finished walls, and rebars inside columns. Roberts et al. (2002) used AR to overlay locations of subsurface electrical, telephone, gas, and water lines onto real-world views. Both applications demonstrated the potential of AR in helping maintenance workers avoid buried infrastructure and structural elements.

Webster et al. (1996) also presented an AR system to guide workers through assembly of a space frame. Hammad et al. (2004) augmented contextual information on real views of bridges to help inspectors conduct inspections more effectively. Thomas et al. (1999) and Klinker et al. (2001) explored AR to visualize designs outdoors. Wang and Dunston (2006) have studied the potential of AR as an assistant viewer for computer-aided drawing. Kamat and El-Tawil (2007) also used AR to study the extent of horizontal displacements sustained by structural elements due to extreme loading conditions.

Graphical Simulation in Outdoor Augmented Reality

On the spectrum between VR, wherein the user is immersed inside a computer generated space and the real world, AR tends to be much closer to the latter. AR integrates images of virtual objects with user views of the real world. In other words, AR allows users to see and navigate in the real world, with virtual objects superimposed on or blended with their view. Fig. 1 presents snap-



Virtual Reality Bridge Construction Model

Augmented Reality Bridge Construction Model

Fig. 1. Comparison of graphical simulation in virtual and augmented reality

shots of a graphical simulation of a bridge construction project in VR and AR. In the VR simulation portrayed on the left, all simulation entities are modeled in virtual space. In contrast, in the AR simulation snapshot depicted on the right, the view of real world and existing structures (i.e., partially completed bridge deck) are used as the background, and CAD models of only the simulation objects under study (i.e., cranes) are superimposed over this view.

AR can be classified into two categories: indoor and outdoor. In indoor AR, the user takes advantage of a prepared and accessible environment. A user's movements are usually restricted to a finite space so that the system can respond faster and with more accuracy, and can generate the augmented view more quickly as the user navigates it. For a domain such as construction, however, indoor AR has limited applications because most construction activities are performed in outdoor, unprepared environments (e.g., heavy construction, roads, bridges, dams, buildings, etc.). Outdoor AR is thus more applicable.

The main requirement of outdoor AR is the need to accurately track the user's viewpoint, and to respond correctly to variations in user movement and environmental characteristics (e.g., irregular terrain, lighting conditions, etc.). In other words, unlike indoor AR where the user is limited to navigating in a restricted space, in outdoor AR the user needs the ability to navigate freely with minimum constraints. At the same time, the AR system must be capable of generating an accurate representation of augmented space in real time so that the user experiences a world in which virtual objects remain fixed to their intended location and seamlessly blend with real entities (Azuma et al. 1997).

Hardware Setup for Mobile Outdoor Augmented Reality

Fig. 2 presents a schematic overview of the hardware components selected for this study's outdoor AR testbed, and the configuration in which they were assembled to effectively achieve the above objectives. A video camera models the behavior of the eye where all the real objects in the surrounding space appear in perspective view (i.e., the viewing area consists of a truncated pyramid in which parallel lines coincide in the horizon). The video camera continually captures a view of the real world and transmits the images to the AR platform's laptop computer. A head-mounted display (HMD) device worn by the user is connected to the computer's video port.

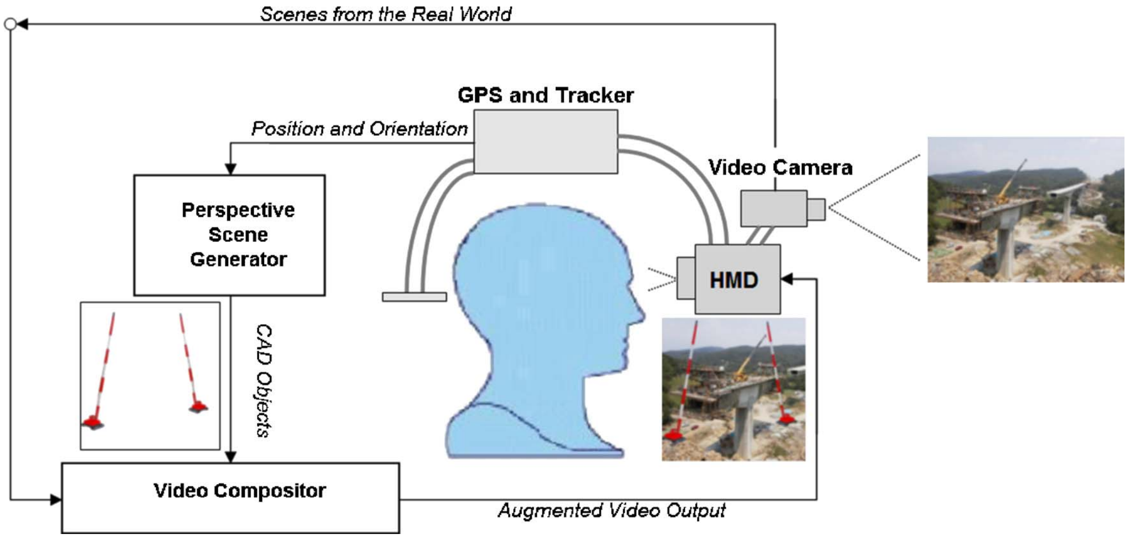


Fig. 2. Hardware setup for mobile outdoor augmented reality

In addition, there are two important pieces of equipment connected to the user to keep track of viewpoint movements and basically provide the input for the registration computations. These are the GPS receiver and a 3-DOF orientation tracker. The GPS receiver provides the AR platform with real-time position data in the global space in the form of longitude, latitude, and altitude (Rogers et al. 1999; Roberts et al. 2002; Dodson et al. 2002). The orientation tracker on the other hand, provides the platform with three important pieces of information regarding the angular motion of the user's head. These are called yaw, pitch, and roll angles with yaw being the rotation about the vertical Y axis, pitch being the rotation about the horizontal X axis, and roll being the rotation about the Z-axis (parallel to the depth of view).

Using these six pieces of information, the AR platform is capable of computing the user's viewpoint in real time, and based on the computed viewpoint position and 3D orientation, can correctly register virtual CAD objects in the user's viewing frustum. The CAD objects are drawn inside a standard open graphics language (OpenGL) perspective viewing frustum. The OpenGL viewing frustum is reconciled with the truncated viewing pyramid of the video camera that captures the real world view. Accurate, real time alignment (i.e., registration) of the two viewing frustums (real and virtual) leads to a realistic augmented view where both real and virtual objects coexist. The AR platform transmits images of this augmented environment to the user's display device.

Technical Approach for Accurate Registration

The most important requirement for realizing an AR scene in which virtual models appear to coexist with objects in the real environment is real-time knowledge of the relationship between the models, real world objects, and the video input device, i.e., video camera (Barfield and Caudell 2001). Registration in AR means accurate overlapping or coinciding of the real and virtual object coordinate frames. Once accurate registration is achieved and maintained, CAD models placed in the augmented space are correctly located and oriented in the real world regardless of where in the augmented space they are viewed from. Registration of virtual objects in the real environment thus requires accurate tracking of the user's viewpoint position and orientation.

To graphically illustrate the problem, consider the snapshots presented in Fig. 3. The first snapshot depicts a real environment on the University of Michigan's north campus. There are a number of real pieces of construction equipment in the background. The objective of this example is to superimpose a CAD model of an excavator over the real background at the location and with the orientation (facing north) indicated on the plan view in the figure's second snapshot. In order to achieve this, the AR system must track the user's movement and head orientation. This is being done repeatedly for every frame. In other words, the local transformation of the user is being calculated at each frame before they are applied to the virtual CAD models in his or her viewing frustum. As a result, the composite augmented view is updated continuously over time with the position and orientation of the virtual CAD models being adjusted based on the user's movements (Behzadan and Kamat 2005a).

The last four snapshots in Fig. 3 depict accurate augmented views an AR user expects to see when looking at the scene from the locations (a, b, c, and d) indicated on the plan view. In each case, the final result is a mixed view of real and virtual objects and is sensitive to the user's location and orientation. When the user stands at point (a) and looks toward the north, the rear end of the virtual excavator is expected to be seen [Fig. 3(a)]. At location (b), the user is standing in front of the virtual excavator looking toward the south. In this case the user must see the excavator from the front [Fig. 3(b)]. When the user is at location (c) looking west, the right side of the excavator must be visible [Fig. 3(c)]. Finally, in location (d) when the user looks east, only the left face of the virtual excavator must be visible in the composite AR output [Fig. 3(d)]. In summary, the augmented virtual excavator must remain fixed to its real-world location while the AR user navigates around it on the jobsite. In this study, this is achieved by using GPS and 3-DOF angular trackers to monitor the user's position and orientation.

Georeferenced Registration of CAD Models Using GPS and 3-DOF Angular Tracking

Fig. 4 depicts the procedural approach used in the presented research to register virtual CAD models in a user's view and superimpose them in the final augmented scene. To perform the registration task, the GPS receiver obtains the user's global position in real time. In addition, the global position of the virtual object (i.e., the excavator in this figure) can be read from a corresponding text file created manually and associated with the CAD model. Knowing these two point locations, the relative distance between the user and the virtual object is calculated. As soon as the distance between the user and the virtual object is determined, the corresponding relative heading (based on the geographical North direction being zero) of the vector connecting the two points is calculated. This is depicted by the symbol α in the figure. By the end of this set of calculations, two important pieces of information about the objects in the user's view are available: the relative distance and the relative angle.

Now, considering the fact that the CAD objects are placed in an OpenGL perspective viewing frustum, which is overlapped with the user's viewpoint frustum, transformation matrices can be effectively used to manipulate them as required. In order to do that, each object is first subjected to a translation matrix by which it is translated into the depth of view by an amount equal to the distance between the two points (R). Then a rotation matrix is applied to the object by which it is rotated about the Y-axis (vertical axis) by an amount equal to the calculated relative angle (α). These two transformations update the virtual object's location based on the last user movement. Fig. 5 is a schematic illustration of the transformation procedure applied to each virtual object inside the user's field of view. The distance and angle computation steps are described in detail in the following subsections.

Computation of Horizontal Distance between User's Viewpoint and Augmented CAD Objects

As the user moves around, the relative distance (R) between the center of the viewpoint and the virtual object (i.e., the excavator in the current example) is calculated using the Vincenty method (Vincenty 1975). This method provides a set of equations to calculate the horizontal component of the distance between two points with known latitude and longitude, and is accurate to 0.5 mm or 0.000015 in. The important characteristic to note regarding these equations is that the computed distance between two points is at mean sea level (MSL) or more exactly along the "ellipsoid datum."

However, it has been shown that if the points in context are located 2 km above the MSL, the distance obtained by the

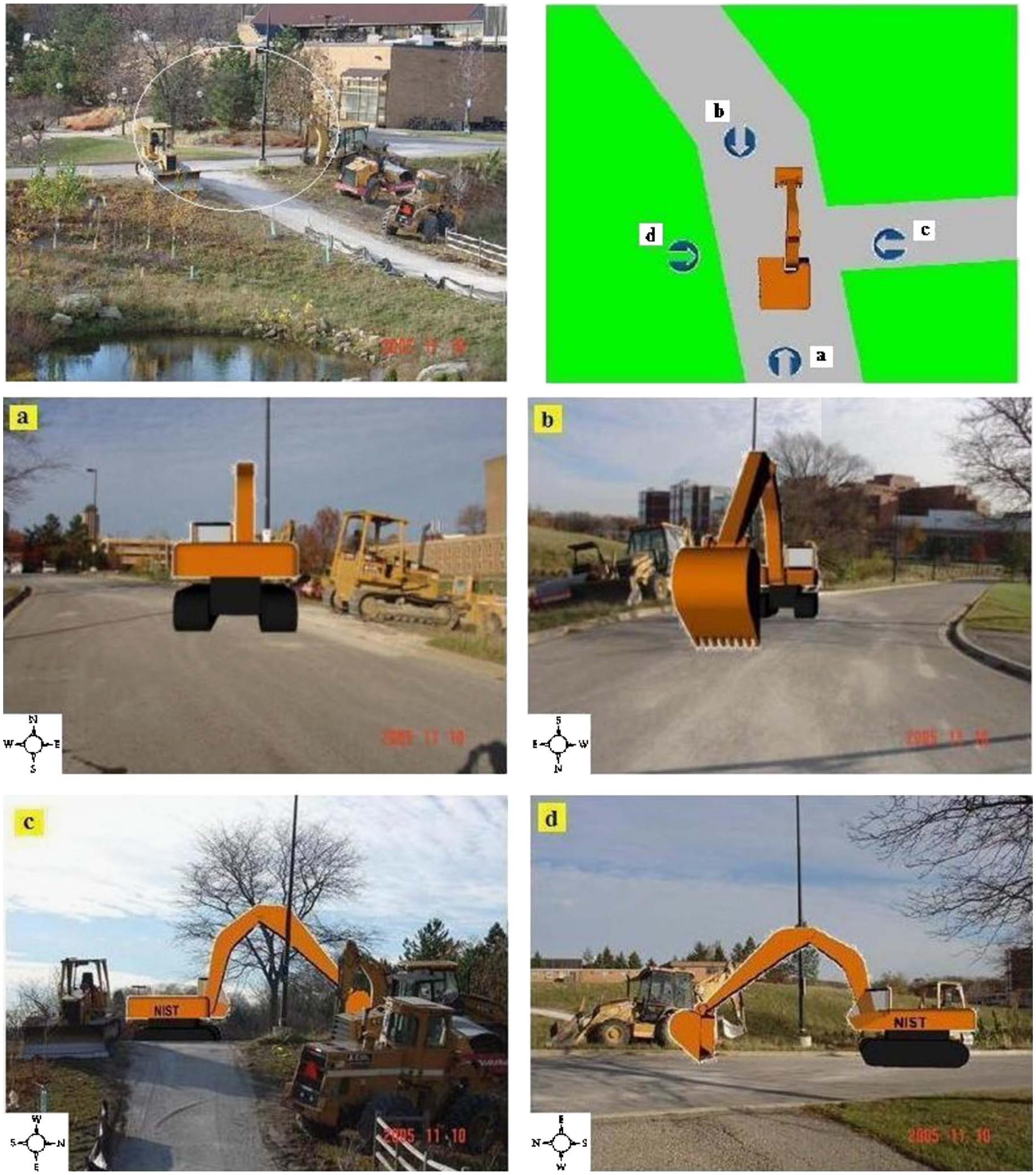


Fig. 3. Concept of accurate outdoor augmented reality registration

Vincenty method will be 0.03% smaller than the actual distance. Thus, for the distance computation in this research, the Vincenty method was found to provide sufficient accuracy for obtaining relative distances between a user and augmented virtual models. Position data of the virtual models is provided to the computation engine via a text file in which the longitude, latitude, altitude, and display time of each object has been previously stored. The content of a sample text file for an AR scene consisting of six virtual models superimposed on a real background is shown in Fig. 6. The applied Vincenty method uses one of the most accurate and widely used models for the earth ellipsoid (i.e., the closest elliptical shape to the actual surface of the Earth in a certain region) in the North American continent, the WGS-84/GRS-80. In this model, the earth ellipsoid is defined using the geometric parameters shown in Fig. 7.

In this figure, (a) and (b) are the major and minor semi-axes of the ellipsoid and (f) is the flattening ratio which is the difference in the length of the two semi axes (i.e., a–b) divided by the length of the major semi axis (i.e., a). The primary variables used in this method are presented in Fig. 8 in the form of a pseudocode. All angles are measured in radians. In addition, the "East" and "North" directions are assumed positive while the "West" and "South" are considered to be negative in sign.

The Vincenty method follows an iterative procedure to calculate the distance between any two given points until it reaches an acceptable level of convergence in which the difference between the last two calculated values of Lambda and Lambdap is less than a specific amount, or until a certain number of iterations have been performed, whatever comes first. Fig. 9 presents the pseudocode for the iterative part of the algorithm.

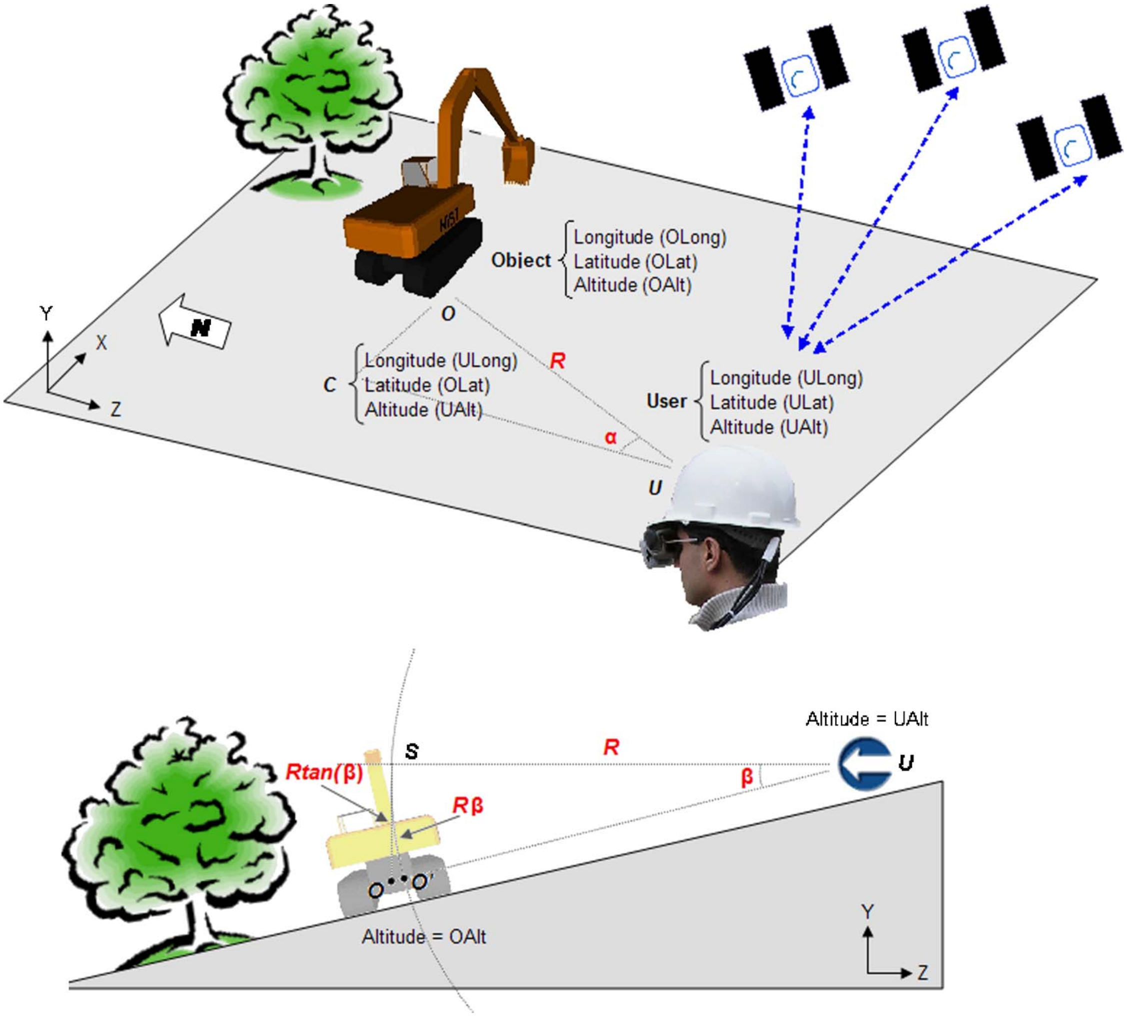


Fig. 4. Technical approach for accurate registration

In the presented code, the error margin has been set at 10^{-12} m. The maximum number of iterations that the loop should execute in trying to reach the desired accuracy is given by the variable, count. The final noniterative segment of the algorithm computes the distance S between the two points with known latitudes and longitudes in meters. The pseudocode for this segment of the algorithm is presented in Fig. 10.

Adjusting Elevation of CAD Objects due to Altitude Differences

Revisiting Fig. 4, the basic assumption in calculating the horizontal displacement (R) and angle (α) is that both the user and the CAD object have the same elevation (i.e., global altitude). Thus, in case the object is located at a higher or lower elevation than the user, further adjustments are needed requiring additional computation steps. In Fig. 4, for example, the object has a lower altitude than the user. In this case, the relative pitch angle between the user and the object (β) must be calculated using properties of triangle USO.

The virtual object is then rotated about the X-axis by an amount equal to the calculated angle β. Referring to Fig. 4, note that by doing this the object is being rotated along the SO' curve. For simplicity, a good assumption can be translating the object along the cord SO in which case the final position of the virtual model ends up to be the point O instead of O'. A final adjustment may be needed to be made on the object's initial side roll angle to represent the possible ground slope (equal to β in Fig. 4). In this study, a pure rotation equal to the slope is being applied to the object around its local X axis so that it lies completely on the ground. In other words, the ground plane is assumed to be uniform and parallel to line segment UO'.

Using the translation along SO instead of rotation along SO' may cause a minute positional error that can be safely neglected for the purposes of this study without experiencing any adverse visual artifacts. However, for wider range applications (e.g., objects that are to be placed far away from the user), the length of OO' can be large in which case, instead of translating the object from S to O, a rotation equal to β must be applied to transform the object from S to O'.

Thus, when a mobile AR user turns his/her head, the relative change in the head orientation is obtained using the three pieces of data coming from the 3-DOF orientation tracker. Applying the reverse transformations in the amount of the computed angles to the virtual objects in the form of a rotation matrix leads to a final augmented view wherein the objects' locations are unaffected by the user's head movement. In addition, in cases where the user both moves and rotates the head at the same time, all the computation steps in the described procedure are continually repeated

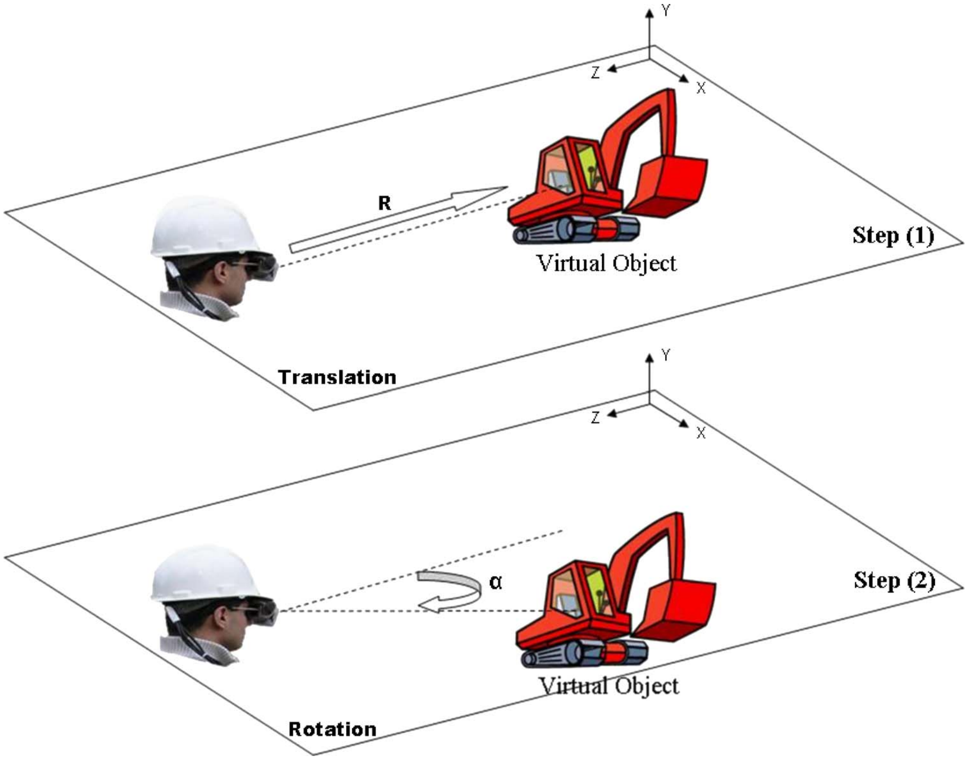


Fig. 5. Transformation procedure to place virtual objects in field of view

(i.e., distance and relative yaw angle calculation, and pitch adjustment) so that the final composite AR view remains unaffected as a user moves freely in the environment.

UM-AR-GPS-ROVER Augmented Reality Platform

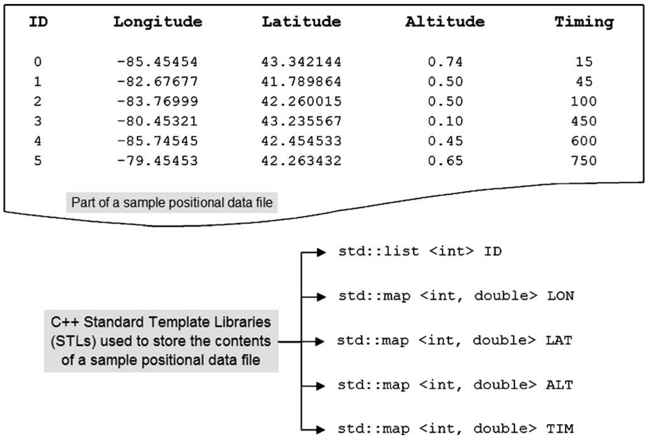
The designed registration algorithms have been implemented in an AR prototype platform named UM-AR-GPS-ROVER that serves as a proof-of-concept, and also helps validate the research results. UM-AR-GPS-ROVER comprises two main components working in parallel. In addition to the software components of the tool, supporting hardware devices are appropriately integrated to provide the necessary sensing, input, and output capabilities. The connected hardware devices primarily consist of a GPS receiver, a 3-DOF orientation tracker, an optional HMD, and a laptop computer.

Description of Hardware Setup

A Delorme Earthmate GPS receiver and an InterSense InterTrax2 head tracking system are used for user position and orientation sensing, respectively. The GPS receiver takes advantage of the wide area augmentation system (WAAS) technology. WAAS enables the GPS receiver to provide positioning data with an accuracy level of better than 3 m 95% of the time (Delorme 2005). This level of accuracy was found to be satisfactory for prototyping purposes. The GPS receiver can also operate in normal range of velocities that a mobile walking or even driving AR user might have. For future prototypes, these peripheral devices are planned to be replaced by more accurate substitutes in order to improve the accuracy of the achieved registration techniques. The writers are currently experimenting with GPS receivers capable of receiving corrections from the Omnistar subscription service and real-time kinematic (RTK) base stations. The former supports a positional accuracy of about 0.1 m while the latter supports an accuracy of as much as 0.01 m. In addition, recent research in GPS quality of service provides an integrated approach that can provide users with predictable positioning solutions that combine GPS availability, quality, and reliability (Karimi et al. 2004).

A Sony DCR-TRV33 mini-DV digital video camera recorder is used as a video capturing device and models the user's viewpoint. The video input device basically captures scenes of the surrounding environment, and provides the background for the user's view. The captured images and the augmented graphics are blended into a composite view and displayed on the laptop computer's screen. The GPS sensor and the orientation tracker are mounted on the camera using Velcro fasteners.

The primary issue in selecting the video capture device is the



```
ID        Longitude       Latitude       Altitude        Timing

0         -85.45454       43.342144        0.74              15
1         -82.67677       41.789864        0.50              45
2         -83.76999       42.260015        0.50             100
3         -80.45321       43.235567        0.10             450
4         -85.74545       42.454533        0.45             600
5         -79.45453       42.263432        0.65             750
```

Part of a sample positional data file

C++ Standard Template Libraries (STLs) used to store the contents of a sample positional data file

```
std::list <int> ID
std::map <int, double> LON
std::map <int, double> LAT
std::map <int, double> ALT
std::map <int, double> TIM
```

Fig. 6. Sample text file of virtual CAD object properties

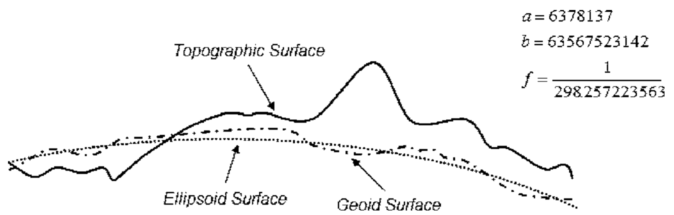


Fig. 7. Basic geometrical parameters used in Vincenty method

capability to simulate the user's viewpoint in terms of both the view depth and resolution. Traditional web cameras, which are good choices for close-distance indoor applications, were found to be unable to focus on objects other than those present within a few feet of the camera. Such cameras were thus found to be inappropriate for outdoor AR environments. Fig. 11 presents photographs of the hardware setup used in UM-AR-GPS-ROVER.

An i-glasses SVGA Pro 3D HMD is used as an optional wearable display. When the HMD is used, the video camera is switched to a Unibrain Fire-i digital firewire camera, and the GPS sensor and orientation tracker are mounted on top of the HMD to create a wearable computing user experience. In wearable computing, for an AR user to be able to navigate freely, the AR platform should be set up in a manner that can be easily carried and accessed anywhere. In the presented work, a HP Pavilion laptop with 3 GHz CPU speed and 512 MB RAM is used. Finding an optimum case between computing power and battery life on one hand, and portability on the other, is an important issue of conflict. That is because the more powerful a computer system is, the more weight it has and hence the more difficulty a user faces carrying it around.

Description of Software Components

Knowing the advantages of modular compared to tightly coupled software components, the main goal of the writers has been to keep the software as modular as possible (Behzadan and Kamat 2005b). The software components of UM-AR-GPS-ROVER are independent interconnected modules that can be easily replaced or updated as necessary. A good example may be replacing an old hardware component (e.g., GPS receiver) with a new one which produces similar output but with more accuracy and reliability. In the present setup, this can be achieved with minimal changes that need to be made, for instance, to account for the new sensors data transmission protocol.

```
// primary variables and constants

$\Delta\mathrm{Long} = \mathrm{Longitude}_2 - \mathrm{Longitude}_1$
$U_1 = \tan^{-1}\left[(1-f)\tan\left(\mathrm{Latitude}_1\right)\right]$
$U_2 = \tan^{-1}\left[(1-f)\tan\left(\mathrm{Latitude}_2\right)\right]$
$\mathrm{SinU}_1 = \mathrm{Sin}\left(U_1\right)$
$\mathrm{SinU}_2 = \mathrm{Sin}\left(U_2\right)$
$\mathrm{CosU}_1 = \mathrm{Cos}\left(U_1\right)$
$\mathrm{CosU}_2 = \mathrm{Cos}\left(U_2\right)$
$\mathrm{Lambda} = \Delta\mathrm{Long}$
$\mathrm{LambdaP} = 2\pi$
$\mathrm{Count} = 0$
```

Fig. 8. Primary geometrical parameters used in Vincenty method

```
// iterative loop to achieve a certain level of accuracy

while$\left(\left|\mathrm{Lambda} - \mathrm{LambdaP}\right| > 10^{-12}\ \mathrm{AND}\ \mathrm{Count} < 100\right)$
{

    $\mathrm{SinLambda} = \mathrm{Sin}\,(\mathrm{Lambda})$
    $\mathrm{CosLambda} = \mathrm{Cos}\,(\mathrm{Lambda})$
    $\mathrm{SinSigma} = \sqrt{\left(\mathrm{CosU}_2 \times \mathrm{SinLambda}\right)^2 + \left(\mathrm{CosU}_1 \times \mathrm{SinU}_2 - \mathrm{SinU}_1 \times \mathrm{CosU}_2 \times \mathrm{CosLambda}\right)^2}$
    $\mathrm{CosSigma} = \mathrm{SinU}_1 \times \mathrm{SinU}_2 + \mathrm{CosU}_1 \times \mathrm{CosU}_2 \times \mathrm{CosLambda}$
    $\mathrm{Sigma} = \tan^{-1}\left(\frac{\mathrm{CosSigma}}{\mathrm{SinSigma}}\right)$
    $\mathrm{Alpha} = \mathrm{Sin}^{-1}\left(\frac{\mathrm{CosU}_1 \times \mathrm{CosU}_2 \times \mathrm{SinLambda}}{\mathrm{SinSigma}}\right)$
    $\mathrm{CosSqAlpha} = \mathrm{Cos}^2\left(\mathrm{Alpha}\right)$
    $\mathrm{Cos2SigmaM} = \mathrm{CosSigma} - 2 \times \frac{\mathrm{SinU}_1 \times \mathrm{SinU}_2}{\mathrm{CosSqAlpha}}$
    $C = \frac{1}{16} \times \mathrm{CosSqAlpha} \times \left(4 + f \times \left(4 - 3 \times \mathrm{CosSqAlpha}\right)\right)$
    $\mathrm{LambdaP} = \mathrm{Lambda}$
    $\mathrm{Dummy} = \mathrm{Cos2SigmaM} + C \times \mathrm{CosSigma} \times \left(-1 + 2 \times \mathrm{Cos2SigmaM}^2\right)$
    $\mathrm{Lambda} = \Delta\mathrm{Long} + (1 - C) \times f \times \mathrm{Sin}\left(\mathrm{Alpha}\right) \times \left(\mathrm{Sigma} + C \times \mathrm{SinSigma} \times \mathrm{Dummy}\right)$
    $\mathrm{Count} = \mathrm{Count} + 1$

}
```

Fig. 9. Iterative computation algorithm used in Vincenty method

The UM-AR-GPS-ROVER platform is implemented as a set of four loosely coupled interacting modules. These are graphically presented in Fig. 12. The first module captures a live video stream from the real environment using the video input device. The second module is mainly a data collector for the GPS receiver and orientation tracker and communicates with the sensors via universal serial bus (USB) ports. This module provides the input for the third module which is the transformation module that is responsible for registering the viewpoint's movements and applying appropriate transformations to the virtual objects in the user's view. The fourth module is essentially a graphical module that reads graphical data from indicated CAD files and places each virtual model in the user's view in real time.

The provided modules are designed considering the requirements of scalability and extensibility. In other words, each module is capable of being upgraded or even replaced by a more efficient module which maintains the minimum required functionalities while providing more accurate and convenient methods to the AR platform. For example, the "GPS and tracker" module is designed in a way that it can establish communication with any corresponding devices as long as the incoming data follow a certain format. Likewise, the "video capture" module is capable of capturing scenes from any video input device that follows the

```
// final computations to obtain the relative distance between two given points

$\mathrm{uSq} = \mathrm{CosSqAlpha} \times \left(\frac{a^2 - b^2}{b^2}\right)$
$A = 1 + \frac{\mathrm{uSq}}{16384} \times \left(4096 + \mathrm{uSq} \times \left(-768 + \mathrm{uSq} \times \left(320 - 175 \times \mathrm{uSq}\right)\right)\right)$
$B = \frac{\mathrm{uSq}}{1024} \times \left(256 + \mathrm{uSq} \times \left(-128 + \mathrm{uSq} \times \left(74 - 47 \times \mathrm{uSq}\right)\right)\right)$
$\mathrm{Temp1} = \frac{B}{6} \times \mathrm{Cos2SigmaM} \times \left(-3 + 4 \times \mathrm{SinSigma}^2\right) \times \left(-3 + 4 \times \mathrm{Cos2SigmaM}^2\right)$
$\mathrm{Temp2} = \mathrm{Cos2SigmaM} + \frac{B}{4} \times \left(\mathrm{CosSigma} \times \left(-1 + 2 \times \mathrm{Cos2SigmaM}^2\right) - \mathrm{Temp1}\right)$
$\mathrm{DeltaSigma} = B \times \mathrm{SinSigma} \times \mathrm{Temp2}$
$S = b \times A \times \left(\mathrm{Sigma} - \mathrm{DeltaSigma}\right)$
```

Fig. 10. Final distance computation between two points using Vincenty method

Fig. 11. Hardware setup of UM-AR-GPS-ROVER

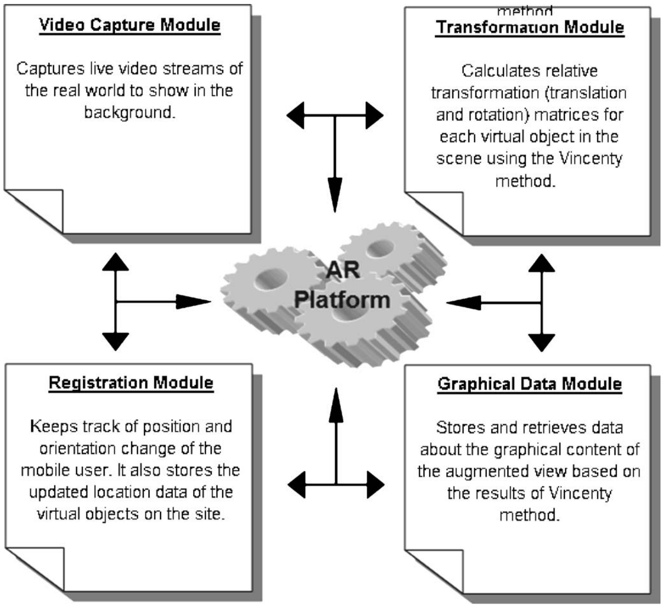


Fig. 12. Interconnection between UM-AR-GPS-ROVER software modules

Microsoft DirectShow format, which is a generic framework developed primarily for client-side streaming (Microsoft 2006).

Using these modules and knowing that the location and orientation of virtual objects should be kept independent of the user's position and orientation, for any transformation of the user, reverse transformations are calculated and applied to the objects in view so they appear fixed to their real world locations.

Validation

To validate the research results and the designed registration algorithms, the implemented UM-AR-GPS-ROVER platform was used to place several static and dynamic three-dimensional CAD models at several known locations in outdoor augmented space (Behzadan and Kamat 2005b). In particular, the prototype was successfully tested in many outdoor locations at the University of Michigan north campus using several 3D and 4D construction models (e.g., buildings, structural frames, pieces of equipment, etc.). Fig. 13 presents two snapshots of structural steel frames registered in outdoor AR.

UM-AR-GPS-ROVER was completely developed in C++ environment using OpenGL functionalities. In order to communicate with the GPS receiver, a C++ library called GPSToolkit was used which provided methods to open the GPS port, receive data from the GPS unit, and extract them in the form of longitude, latitude, and altitude. To establish communication with the 3-DOF orientation tracker, the sample code provided by the device manufacturer was modified and used inside the AR application.

UM-AR-GPS-ROVER supports CAD objects created in the virtual reality modeling language (VRML) format since it provides several features and functionalities to create a scene description of 3D interactive models. VRML models can be created in any external CAD program (e.g., 3D Studio MAX, or Auto CAD) and exported to other graphical applications (Kamat and Martinez 2001). Another useful feature of VRML is that the initial scale and orientation of each of the CAD models created using VRML can be modified when they are loaded in an external application. This feature is widely used in this study so that the dimensions and orientation of the loaded models comply with the global coordinate frame properties of the real environment.

In order to test the prototype's ability to augment dynamically changing graphical models in a user's view, a 4D CAD (Koo and Fischer 2000) model of a structural steel frame was registered at a known outdoor location. In particular, scheduled construction activities for the erection of the steel frame (columns, beams, girders, and connection) were graphically animated with the passage of simulated project time. The simulated project time is introduced as an extra dimension for each augmented virtual model such that an object (e.g., steel beam or column) is not superimposed in the augmented view unless the simulation clock passes its scheduled completion time.

In order to allow UM-AR-GPS-ROVER to identify the appropriate "appearance time" for each CAD object, time tags for all virtual models were stored in a text file with their corresponding model ID number. At each frame in the graphical AR simulation, the text file is searched to determine whether there are object(s) with a time tag smaller than the current simulation time. If an object falls into this category, its graphical data are read from a corresponding graphical file and are displayed in the user's view at the correct expected location. Fig. 14 presents snapshots of the 4D graphical AR simulation.

To validate the capability of the presented platform to perform accurate registration, global position of the CAD models shown in Figs. 11 and 12 were obtained on site and input to the system via the associated text file. The user's position and orientation were also tracked by a GPS receiver and a 3-DOF orientation tracker while he was moving on the site. The implemented Vincenty algorithm calculated real time relative distance between the user and the CAD objects and the final view

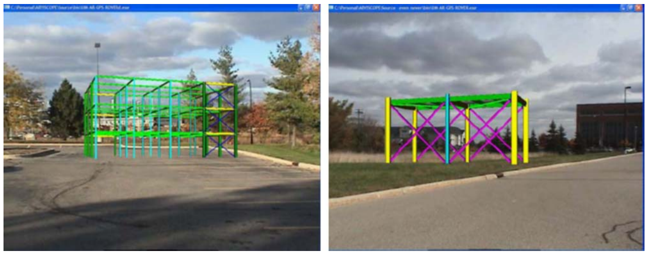


Fig. 13. Structural steel frames registered in outdoor augmented reality

was updated accordingly to produce a realistic augmented view based on the results of the registration technique and Vincenty algorithm.

As discussed earlier, the accuracy of the acquired positioning data from the GPS receiver is always a major concern. However, in the validation stage of the present platform and in order to prove the feasibility of the developed methods and algorithms in this study, the writers decided to choose WAAS since it provides a free GPS signal correction service. However, since the methods and functionalities developed in this study are designed in compliance with National Marine and Electronics Association (NMEA) which is a widely accepted standard for GPS data transmission, future possible upgrades to higher accuracy devices will not affect the application. This is mainly because the issue of accuracy is more internal to the incoming data streams from the GPS device and as long as these data streams follow the NMEA format, the AR application developed in this study is capable of receiving and extracting them in the form of useful position values.

Considering all these factors, in choosing the location of the outdoor tests at this stage and considering the geographical layout of the state of Michigan, the writers tried to perform the tests in areas in which the south sky horizon was clear enough and not blocked by either natural (e.g., trees) or human-made (e.g., buildings) objects for the WAAS satellite to be viewed by the GPS receiver. This in fact led to an increased level of accuracy in the validation stage. To be more specific, the overall accuracy level achieved in the outdoor test was consistently about 1 m which is an acceptable range considering that the GPS receiver used in these tests was not a high-accuracy device.

Future Work and Challenges

Interesting problems for future research primarily fall under three categories: accuracy, visual realism, and ergonomics. Increased accuracy in wide area tracking is an important issue. An outdoor AR system should provide the user with the freedom to operate in a wide area and at the same time keep accurate track of the user's position and orientation. In the presented research, the level of accuracy corresponding to each piece of data obtained from the orientation tracker is based on the quality and accuracy of the previously obtained data point. In other words, the tracker provides relative changes in orientation along the three primary axes.

Thus, phenomena such as rapid angular movements or fluctuations in ambient temperature induce a significant amount of drift in the orientation tracker's readings, making the tracked orientation unreliable. The same is true of the GPS receiver. In locations where the WAAS satellite is visible, the receiver's reported position was typically found to be accurate to within 1 m or better.

Fig. 14. Graphical augmented reality simulation of a 4D CAD model

However, an interruption in the WAAS signal severely degrades the quality of the reported GPS position. In order to address these issues, the writers are currently experimenting with a high-accuracy three-axis tilt compensated compass module for orientation tracking and an Omnistar-HP differential correction capable GPS receiver for position sensing. The compass data at each instance reports the absolute magnetic heading (yaw), roll, and pitch, and is thus expected to be free from drift issues. On the other hand, Omnistar-HP corrected GPS locations are documented to be accurate within 0.1 m of the real positions.

Another important problem in outdoor AR is that of occlusion. Incorrect occlusion happens when a real object is placed between the user's view and the virtual object(s) in augmented space. In such a situation, as the distance between the real object and the user is less than that between the virtual object(s) and the user, the real object should theoretically block (at least partially) the user's view of the superimposed models that lie behind it. This idea is conceptually presented in Fig. 15.

The stick of the virtual excavator in this figure is partially occluded at two locations by the light pole and the stick of a real excavator, both of which are closer to the user in augmented space than the virtual excavator. Thus, under ideal circumstances, the hidden portions of virtual models should not be visible in the composite AR output as shown in the figure. That is, however, not currently the case because UM-AR-GPS-ROVER draws the pixels of all virtual models after painting the captured video image as a background.

A possible solution to address this issue could use a combination of rapid geometric modeling of the surrounding environment or other depth sensing techniques (e.g., stereo cameras) and utilize the graphics processor's z buffer to draw the appropriate set of pixels in each composite AR frame. In other words, if this depth of real objects is greater than the depth of virtual object(s) for a given view, the real object does not occlude any virtual objects. In the opposite set of circumstances, appropriate corrections should be made to the user's view to take into account the existence of an occluding real object.

Fig. 15. Dynamic occlusion problem in outdoor augmented reality

Finally, the effects of and possible limitations caused by the use of wearable computer packages on AR users presents important challenges in ergonomics and human issues (Barfield and Caudell 2001). While a user expects real time response from the AR platform during navigation, factors such as computational time delays or positional errors affect performance. On the other hand, having a computer system with a high computational power in turn increases the overall weight of the equipment to be carried by the AR user, and at the same time reduces flexibility. Finding an optimum solution that fulfills both ergonomic and technical requirements in a mobile outdoor AR platform is a challenging prospect that must be closely studied.

Summary and Conclusions

The primary advantage of graphical simulation in AR compared to that in VR is the significant reduction in the amount of effort required for CAD model engineering. AR superimposes graphical images of virtual objects on to user views of the real surrounding space, thereby precluding the need to CAD model the entire simulation environment. In order for AR graphical simulations to be realistic and convincing, real objects and augmented virtual models must be properly aligned relative to each other. Without accurate registration, the illusion that the two coexist in AR space is compromised.

It was found that tracking systems traditionally used for AR registration are intended for use in controlled indoor spaces and are unsuitable for unprepared outdoor environments such as those found on typical construction sites. In order to address this issue in the presented research, the global outdoor position and 3D orientation of the user's viewpoint (i.e., longitude, latitude, altitude, heading, pitch, and roll) are tracked using a GPS sensor and a 3-DOF orientation sensor. The tracked information is reconciled with the known global position and orientation of CAD objects to be superimposed in a user's view.

Based on this computation, the relative translation and axial rotations between the user's eyes and the CAD objects are calculated at each frame during visualization. The relative geometric transformations are then applied to the CAD objects to generate an augmented outdoor environment where superimposed CAD objects remain fixed to their real world locations as the user moves about freely on a construction site. Validation of the designed algorithms with 3D and 4D construction models proved that this approach is not only feasible, but also very effective.

It was found that the choice of a differential orientation tracker that measures relative values of horizontal rotation (yaw) instead of global magnetic or true heading is not a good choice for a mobile AR platform due to issues of accumulating drift. In addition, due to the WAAS satellite being very close to the southern horizon when sighted from southeast Michigan, it was found that the line of sight was frequently lost during navigation, introducing a significant error in the recorded GPS positions. The writers' ongoing research is attempting to address these issues by investigating the use of tilt compensated digital compass modules for orientation tracking and commercial differentially corrected GPS position sensors for accurate location tracking.

The presented UM-AR-GPS-ROVER prototype is the result of the first stage of this ongoing research and is capable of providing an accurate and realistic mixed view of the simulated construction environment with all the required graphical features as well as live scenes of the real world in which the user has the capability of interacting with the environment and observing the immediate results of the interaction.

Acknowledgments

The presented work has been supported by the National Science Foundation (NSF) through Grant No. CMS-0448762. The writers gratefully acknowledge NSF's support. The writers would also like to thank Mr. Robert R. Lipman of the National Institute of Standards and Technology (NIST) for providing the CAD models of steel frames used in the validation exercises of this study. Any opinions, findings, conclusions, and recommendations expressed in this paper are those of the writers and do not necessarily reflect the views of the NSF or the individual acknowledged above.

References

Azuma, R. (1997). "A survey of augmented reality." *Teleoperators and Virtual Environments*, 6(4), 355–385.

Azuma, R., Baillot, W., Behringer, R., Feiner, S., Julier, S., and McIntyre, B. (2001). "Recent advances in augmented reality." *IEEE Comput. Graphics Appl.*, 21(6), 34–47.

Bachmann, E. R., McGhee, R. B., Yun, X., and Zyda, M. J. (2001). "Inertial and magnetic posture tracking for inserting humans into networked virtual environments." *VRST 01*, Banff, Alberta, Canada.

Barfield, W., and Caudell, T., eds. (2001). *Fundamentals of wearable computers and augmented reality*, Lawrence Erlbaum Associates, Mahwah, N.J.

Behzadan, A. H., and Kamat, V. R. (2005a). "UM-AR-GPS-ROVER augmented reality platform: Visualization of construction graphics in outdoor augmented reality." *UMCEE Technical Rep. No. 05-09*, Dept. of Civil and Environmental Engineering, Univ. of Michigan, Ann Arbor, Mich.

Behzadan, A. H., and Kamat, V. R. (2005b). "Visualization of construction graphics in outdoor augmented reality." *Proc., Winter Simulation Conf.* (WSC05), Orlando, Fla.

Brooks, F. P., Jr. (1999). "What's real about virtual reality?" *IEEE Comput. Graphics Appl.*, 19(6), 16–27, ⟨http://www.cs.unc.edu/~brooks⟩ (September 2005).

Delorme (2005). "Earthmate GPS LT-20." ⟨http://www.delorme.com/earthmatelt20⟩ (December 2005).

Dharwadkar, P. V., Varghese, K., O'Connor, J. T., and Gatton, T. M. (1994). "Graphical visualization for planning heavy lifts." *Proc., 1st Congress on Computing in Civil Engineering*, ASCE, Reston, Va.

Dodson, A. H., Roberts, G. W., and Ogundipe, O. (2002). "Construction plant control using RTK GPS." *Proc., FIG 22nd Int. Congress*, Washington, D.C.

Hammad, A., Garrett, J. H., and Karimi, H. (2004). "Location-based computing for infrastructure field tasks." *Telegeoinformatics: Location-based computing and services*, CRC, Boca Raton, Fla., 287–314.

Kamat, V. R., and El-Tawil, S. (2007). "Evaluation of augmented reality for rapid assessment of earthquake-induced building damage." *J. Comput. Civ. Eng.*, in press.

Kamat, V. R., and Martinez, J. C. (2001). "Visualizing simulated construction operations in 3D." *J. Comput. Civ. Eng.*, 15(4), 329–337.

Kamat, V. R., and Martinez, J. C. (2005). "Dynamic 3D visualization of articulated construction equipment." *J. Comput. Civ. Eng.*, 19(4), 356–358.

Karimi, H. A., Liu, X., Liu, S., and Hammad, A. (2004). "GPSLoc: Framework for predicting global positioning system quality of service." *J. Comput. Civ. Eng.*, 18(3), 196–206.

Klinker, G., Stricker, D., and Reiners, D. (2001). "Augmented reality for exterior construction applications." *Fundamentals of wearable computers and augmented reality*, W. Barfield and T. Caudell, eds., Lawrence Erlbaum, Mahwah, N.J.

Koller, D., Klinker, G., Rose, E., Breen, D., Whitaker, R., and Tuceryan, M. (1997). "Real-time vision-based camera tracking for augmented reality applications." *Proc., VRST 97*, Lausanne, Switzerland.

Koo, B., and Fischer, M. (2000). "Feasibility study of 4D CAD in commercial construction." *J. Constr. Eng. Manage.*, 126(4), 251–260.

Lawson, S. W., and Pretlove, J. R. G. (1998). "Augmented reality for underground pipe inspection and maintenance." *Proc., SPIE Int. Symp. on Intelligent Systems and Advanced Manufacturing, Telemanipulator and Telepresence Technologies V*, Boston.

Livingston, M. A., Rosenblum, L. J., Julier, S. J., Brown, D., Baillot, Y., Swan, J. E., II, Gabbard, J. L., and Hix, D. (2002). "An augmented reality system for military operations in urban terrain." *Interservice/Industry Training, Simulation, and Education Conf.*, Orlando, Fla., 1–8.

Microsoft (2006). "What is Microsoft DirectShow?" ⟨http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwmt/html/ds_wm_faq.asp⟩ (June 2006).

Min, P., Halderman, J. A., Kazhdan, M., and Funkhouser, T. A. (2003). "Early experiences with a 3D model search engine." *Proc., 8th Int. Conf. on 3D Web Technology*, ACM, New York, 7–18.

Op den Bosch, A. (1994). "Design/construction processes simulation in real-time object-oriented environments." Ph.D. dissertation, Georgia Institute of Technology, Atlanta.

Piekarski, W., and Thomas, B. H. (2002). "The Tinmith system: Demonstrating new techniques for mobile augmented reality modeling." *Proc., 3rd Australasian User Interfaces Conf.*, Melbourne, Australia.

Roberts, G. W., Evans, A., Dodson, A., Denby, B., Cooper, S., and Hollands, R. (2002). "The use of augmented reality, GPS, and INS for subsurface data visualization." *Proc., FIG 22nd Int. Congress*, Washington, D.C.

Rogers, S., Langley, P., and Wilson, C. (1999). "Mining GPS data to augment road models." *Proc., 5th Int. Conf. on Knowledge Discovery and Data Mining*, San Diego, 104–113.

Thomas, B., Piekarski, W., and Gunther, B. (1999). "Using augmented reality to visualize architecture designs in an outdoor environment." *Proc., Design Computing on the Net (DCNET1999)*, Sydney, Australia.

Vincenty, T. (1975). "Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations." *Surv. Rev.*, 176, 88–93.

Wang, X., and Dunston, P. S. (2006). "Potential of augmented reality as an assistant viewer for computer-aided drawing." *J. Comput. Civ. Eng.*, 20(4), 437–441.

Webster, A., Feiner, S., MacIntyre, B., Massie, W., and Krueger, T. (1996). "Augmented reality in architectural construction, inspection, and renovation." *Proc., Computing in Civil Engineering*, ASCE, Reston, Va, 913–919.

Welch, G., Bishop, G., Vicci, L., Brumback, S., Keller, K., and Colucci, D. (1999). "The HiBall tracker: High performance wide area tracking for virtual and augmented environments." *Proc., VRST 99*, London.

Zhang, H., Shi, J. J., and Tam, C. M. (2002). "Iconic animation for activity-based construction simulation." *J. Comput. Civ. Eng.*, 16(3), 157–164.