Engineering with Computers
DOI 10.1007/s00366-013-0323-0

ORIGINAL ARTICLE

Feasibility of real-time graphical simulation for active monitoring of visibility-constrained construction processes

Sanat Talmaki · Vineet R. Kamat · Kamel Saidi

Received: 10 January 2013 / Accepted: 6 June 2013

Abstract The lack of clear visibility and spatial awareness frequently results in construction accidents such as workers being struck by heavy equipment; and collisions between equipment and workers or between two pieces of equipment. In addition, certain processes such as excavation and drilling inherently pose constraints on equipment operators' abilities to clearly perceive and analyze their working environment. In this paper, the authors investigate the types of spatial interactions on construction sites and the need for graphical real-time monitoring. A computing framework is presented for monitoring interactions between mobile construction equipment and static job-site entities, workers, and other equipment. The framework is based on the use of sensor-based tracking, georeferenced models, and a resulting concurrent, evolving 3D graphical database. The developed framework enables a real-time 3D visualization scheme that provides equipment operators with graphical job-site views that are not possible through conventional on-site cameras. The two key parameters affecting a proximity monitoring framework's effectiveness are measurement error and latency. Measurement error refers to the error in proximity computation—with respect to ground truth or theoretically expected values. Latency is a difference in the time between when an event occurs in the real world and when a proximity monitoring framework provides output to warning systems that end users depend upon. Results from validation experiments conducted to analyze the achievable measurement error and latency of the monitoring framework using indoor GPS tracking as a ground truth system are also presented and discussed.

Keywords Accidents · Buried utilities · Proximity queries · Collision avoidance · Simulation · Monitoring · Sensor-based tracking · Real-time visualization

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

S. Talmaki · V. R. Kamat (✉)
Department of Civil and Environmental Engineering, University of Michigan, 2340 G.G. Brown, 2350 Hayward, Ann Arbor, MI 48109, USA
e-mail: vkamat@umich.edu

S. Talmaki
e-mail: stalmaki@umich.edu

K. Saidi
National Institute of Standards and Technology, Gaithersburg, MD 20899, USA
e-mail: kamel.saidi@nist.gov

1 Introduction

Civil engineering projects are unique as they lead to the creation of an unstructured, dynamic, and continuously evolving work space [50]. Projects in crowded urban areas present project participants with narrow, constrained work spaces and limited visibility of resources that increase the probability of collisions between equipment, workers, materials, and jobsite infrastructure [8, 56]. Projects involving the use of large equipment increase the risk of their collision with workers due to reduced spatial awareness for operators as a result of blind spots on equipment and haul roads. Some projects require equipment to be operated remotely through a process called teleoperation in which operators control equipment with the aid of video

Published online: 21 June 2013

cameras and wireless technology. Such operations introduce an additional obstacle in achieving spatial awareness due to limited field of view afforded by on-board cameras [7].

The lack of clear visibility and spatial awareness results in accidents such as workers-on-foot being struck by heavy equipment; and collisions between equipment and workers or two pieces of equipment [8]. In addition to the above scenarios, certain projects inherently pose constraints on equipment operators' ability to clearly perceive and analyze their working environment [61]. For example, excavator operators carrying out excavation operations in presence of underground utilities are faced with the constant risk of striking buried utilities. Operators must rely on judgment and experience to determine the position of buried utilities in the absence of equipment and infrastructure tracking. Another example of the lack of clear visibility is the case of drilling equipment operators carrying out operations on reinforced concrete slabs. As the underlying reinforcement steel and utility conduits are hidden from the operators' view, they cannot be certain of correct drilling locations in the absence of drill-bit position–orientation tracking with respect to the obstruction locations.

In 2009, visibility-related fatalities that included equipment accounted for a total of 521 fatalities due to workers being struck by moving equipment [21]. Excavation-related damage to buried utilities caused 544 major accidents resulting in 37 fatalities, 152 injuries, and close to $200 million in property damage [45]. The high number of accidents resulting in injuries, fatalities, and monetary and productivity loss has thus led to significant research interest in tackling the problem of construction jobsites accidents that are related to reduced spatial awareness.

In this paper, the authors investigate the types of occurring spatial interactions and the need for real-time monitoring on construction jobsites. A computing framework is presented for monitoring interactions between mobile construction equipment and other job-site entities such as buried infrastructure, workers-on-foot, and other equipment. The framework is based on the use of sensor-based tracking and an evolving 3D graphical database. Results from the experiments conducted to analyze the achievable measurement error of the monitoring framework using indoor GPS tracking as a ground truth system are presented and discussed. The remainder of this paper is structured as follows: Sect. 2 provides an overview of equipment-related accidents on construction job sites and currently adopted avoidance technologies and methodologies. Sections 3 and 4 present limitations with current approaches and the authors' proposed methodology. Section 5 introduces the technical approach and Sect. 6 presents an implemented framework called PROTOCOL that is designed for real-time spatial queries in 3D visualizations of concurrent engineering processes. The results from validation experiments are provided in Sect. 7. Limitations and future improvements are presented in Sect. 8.

2 Literature review

Due to the large number of accidents and fatalities on construction jobsites, significant prior research has been conducted to understand their root causes. While the US construction industry accounts for only around 7 % of the total workforce, it is responsible for nearly 20 % of all industrial fatalities [34]. The common construction jobsite accidents are workers being struck by equipment or falling objects, workers falling into trenches or openings, electrocution, burns, and incidents involving collapse of structures.

Table 1 shows the number of fatalities caused due to workers being struck by an object or equipment in construction, mining, and manufacturing industries over a 5-year period as recorded by the Bureau of Labor Statistics (BLS, [6]).

It is estimated that approximately one-fourth of the construction worker deaths are the result of collisions, rollovers, struck-by accidents, and a variety of other equipment-related incidents [21]. One of the chief contributing factors is workers-on-foot and large mobile equipment sharing the same workspace. Appreciation of the risks involved has resulted in rules and policies governing the operation of equipment on construction jobsites. Occupational Safety and Health Administration (OSHA, [43]) regulations 29 CFR 1926 Subpart O, 1926.601(b)(4) and 1926.602(a)(9) require mobile construction equipment to be equipped with operational back-up alarms and/or a spotter that signals when the equipment can safely backup. There is a requirement for the alarms to be audible even in the presence of high ambient noise that is typical on a construction jobsite [43]. These regulations further reinforce the fact that equipment operators suffer from reduced visibility and spatial awareness.

Table 1 Fatalities per year due to workers being struck by objects or equipment in construction, mining, and manufacturing sectors [6]

Years	Construction	Mining	Manufacturing
2010	141	41	93
2009	154	34	95
2008	208	51	116
2007	215	71	138
2006	223	60	152

The situation is more severe during night-time construction due to the lack of natural light and poorly lit workspaces. A large percentage of highway construction takes place at night to reduce its adverse effects on traffic flow [2]. Thus, the lack of visibility becomes a greater contributing factor. Poor lighting and visibility-related issues were found to be one of the leading causes of accidents involving traffic and workers, and construction equipment and workers [2].

The following statistics on worker safety from the Federal Highway Administration (FHWA) and BLS show that lack of clear visibility and spatial awareness is a major cause of a large number of accidents. Nearly half of the worker fatalities are caused when workers are run over or backed over by vehicles or mobile equipment. Between 2005 and 2010 runovers/backovers were the cause of an average of 48 % of worker fatalities. The second most common cause of worker fatalities is collisions between vehicles/mobile equipment. The third most common cause of worker fatalities is workers caught between or struck by construction equipment and objects. Between 2005 and 2010, this was the cause of an average of 14 % of worker fatalities [15].

Operations that involve concealed or buried infrastructure such as excavation, trenching, and drilling also pose visibility and spatial constraints on operators. The lack of knowledge of the equipment end effectors' location in relation to concealed infrastructure results in inadvertent strikes. Thus, there is a clear need to supplement workers and equipment operators' visibility through additional means and provide increased spatial awareness. Prior research addressing these issues can be generally classified into: (1) vision-based environment recreation approaches and (2) position–orientation tracking-based approaches as summarized in Table 2.

Several technologies have been used to improve positional awareness and knowledge of the surroundings (e.g., [54, 55]). A preliminary real-time crash avoidance framework using 3D imaging sensors for equipment control was presented by Chi et al. [10]. Kim et al. [27] demonstrated the use of a laser rangefinder along with pan and tilt kinematics to recreate the job site using convex hulls and workspace partitioning. Simulations involving obstacle avoidance, artificial-potential function, and minimum distance algorithms showed the potential of a job-site warning system in preventing collisions [27]. Sparse range point clouds using laser range finders have also been used to produce a rapid 3D modeling approach with <5 % error for object recognition [29]. Cho et al. [11] presented a new method for rapid modeling and visualization of a local area based on geometric information about objects obtained using simple sensors (such as a single-axis laser range-finder and a video camera) for better planning and control of construction equipment operations in unstructured workspaces. Son et al. [50, 51] presented the use of flash LADARs for real-time 3D modeling of a construction worksite for autonomous heavy equipment operation.

Radio Frequency (RF) technology has been implemented to provide real-time warnings to RF-tagged equipment operators and workers by triggering alarms when the distance between equipment and worker drops below a safety threshold [57]. Low-frequency, low-power magnetic fields-based radio frequency identification (RFID) technology was employed by Schiffbauer and Mowrey [48] to provide warnings when workers and heavy equipment come too close to each other. Oloufa et al. [44] have demonstrated the use of global positioning system (GPS)-based tracking for providing warnings about impending collisions between equipment by calculating the intersection point of the two vectors representing two moving vehicles. We refer the reader to Zhang et al. [61] for a comprehensive and informative survey and references to other work using GPS for equipment tracking and automated control of construction equipment. Hwang [24] and Cheng et al. [9] also demonstrated the use of ultra-wide band (UWB) technology to monitor dynamic construction resources operating in harsh outdoor environments. Ruff and Hession-Kunz [47], Teizer et al. [57], and Marks and Teizer [36] implemented RFID technology to develop warning systems that monitor worker proximity to heavy construction equipment.

3 Limitations of existing methodologies

Any collision detection or proximity monitoring algorithm between equipment, personnel, and/or job-site infrastructure involves at least two entities. The entities can be static, such as on-site materials and job-site infrastructure, or dynamic, such as equipment and workers-on-foot. It is

Table 2 Existing implementations and technologies can be divided into sensor-based approaches and vision-based approaches

Vision-based job-site reconstruction	Position–orientation tracking sensor
Laser scanning—sparse point clouds	Global positioning system (GPS)
Laser scanning—dense point clouds	Radio frequency identification (RFID)
Flash laser detection and ranging (LADAR)	Radio frequency (RF) sensing and actuating
Computer vision—object recognition	Ultra-wide band (UWB)

Table 3 Types of interactions between entities on a construction job site (assumes that equipment and its operator are viewed as a single entity)

Entity I	Entity II
Equipment	Equipment
Equipment	Worker-on-foot
Equipment	Job site infrastructure
Equipment	Job-site materials
Worker-on-foot	Job-site infrastructure
Worker-on-foot	Job-site materials
Job-site materials	Job-site infrastructure

required that a safety monitoring system on a busy job site be capable of tracking distance on a one-to-many basis. Table 3 summarizes the possible interactions between various entities present on a typical construction job site.

The existing collision detection or proximity monitoring methodologies can be divided into two groups. The first group consists of those technologies in which tracking sensors can be placed on entities I and II from Table 3. Technologies such as GPS, RF transmission, UWB, and RFID, all require sensors to be present on both entities to obtain their location for use in proximity analyses. The second group of technologies includes those that re-create job-site infrastructure and layout in 3D models for use in real-time analysis. Thus, there is a requirement for entities such as materials and infrastructure to be in the line-of-sight of these sensing technologies (such as 3D imaging sensors).

The unique and diverse nature of construction job sites means that no single technology can be applied to all scenarios owing to physical and practical constraints. For example, underwater construction will pose limitations on the use of GPS and RF transmission for position tracking. Similarly, operations where infrastructure is covered by soil and/or concrete will limit the efficacy of vision-based sensing technologies. Object modeling and/or recognition requires that the object be visible to laser scanners and cameras. This is not possible for rebar embedded in concrete and buried utility.

Operations such as these require knowledge of the location of the dynamic entity (excavator, drilling equipment) as well as the embedded entity. Tracking the former entity is possible through the use of tracking sensors such as GPS. However, the latter entity involved in the query is neither capable of being instrumented with sensors nor visible to vision-based technologies such as laser scanners. As a result, the existing methodologies cannot be directly applied to such operations. A new methodology combining tracking sensors instrumented on the dynamic entity (entity I) and a graphical database representing the static entity (entity II) is thus developed and proposed to assist equipment operators in gathering the necessary spatial awareness. The following section describes the proposed methodology using construction operations involving underground or out-of-sight infrastructure as a motivating example.

4 Proposed methodology

Operations such as excavation in the presence of buried utilities pose additional challenges to equipment operators. Lack of accurate information about the location of the hidden infrastructure in elevation (vertical plane) and plan (horizontal plane) reduces the spatial awareness of an operator with respect to their work zone. For example, excavating beyond the permissible safe depth in the presence of buried utilities can trigger accidents, leading to loss of life and damage to equipment and property [45]. To address this situation, a methodology involving the combination of sensor-based tracking and concurrent analysis of a graphical database is proposed. While the developed methodology can be applied to any operation involving hidden infrastructure, the example of buried utility excavation is used herein for illustration purposes.

An excavator operator digging in the presence of subsurface utilities (as seen in Fig. 1) is primarily concerned about inadvertent contact with underlying pipes, cables, and conduits. To be able to dig safely and efficiently, an operator must know the location of the excavator's digging implement relative to buried utilities lying in close proximity. An effective means to convey this information to the operator is visually. Furthermore, the presence of audio-visual warnings to inform the operator of impending collision between a digging implement and underlying utilities can help prevent potential accidents.

The basic proximity monitoring relationship is between an excavator (specifically its digging implement) and a single buried utility. Warnings and real-time 3D visualization are achieved by (1) instrumenting the dynamic entity in the relationship with tracking sensors (for example—

Fig. 1 Limited visibility facing excavator operators and uncertainty regarding location of buried utilities

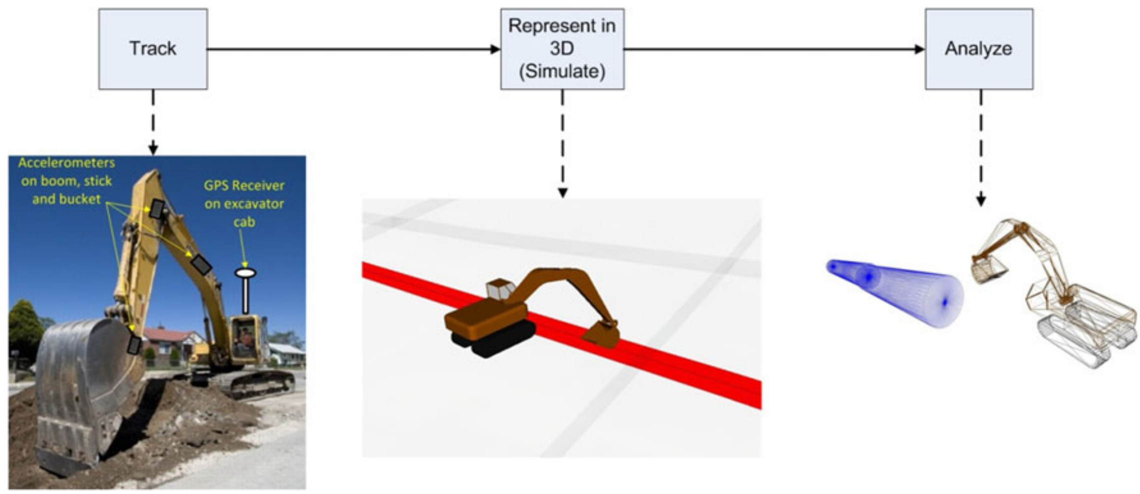

Fig. 2 Proposed methodology: track equipment on job site, represent in real-time 3D simulation, analyze 3D graphical database for proximity and collisions

instrumenting an excavator with position and tilt sensors); (2) creating a 3D graphical database to represent the static and dynamic entities and (3) analyzing interactions between entities in a real-time 3D visualization. Figure 2 illustrates the proposed methodology. The dynamic entity's 3D model is updated in real-time through tracking sensors at the job site. Thus, the 3D visualization serves as a simulation of the real-world operation. As the operation proceeds, the graphical database is analyzed for proximity as well as collisions between included entities. Results from the analysis can be used to trigger audio-visual warnings to notify the operator with vital information such as (1) proximity of excavator's digging implement to underlying utility; (2) breach of safety threshold; and (3) impending collision. The post-analysis reaction is tailored to suit the requirements of a specific operation.

The ability to view operations in real-time improves the overall awareness of operators and site managers. Research indicates that construction operations such as earth moving, heavy lifting, material handling, and remote excavation in cofferdams can be performed more safely and effectively using graphical models of both the equipment and the workspace [11, 22, 29]. Thus, in addition to audio-visual warnings for the operator, real-time 3D visualization of the operation is also an essential component of the proposed methodology.

5 Technical approach

The previous section introduced the proposed methodology to enable tracking for operations involving buried infrastructure. The methodology, as shown in Fig. 2, consists of three stages: track, represent in 3D, and analyze. The technical approach is applied to an operation involving a dynamic entity such as an excavator and one or many underground utility lines. However, the same approach can be applied to any operation involving a pair of dynamic entities or static entities such as material or job-site infrastructure as given in Table 3. This approach requires real-time position–orientation tracking of dynamic entities and 3D geometric models representing the dimensions and positions of static entities. The final stage is completed by analyzing the geometric database consisting of all geometric entities involved in the operation.

Efficiency, accuracy, interactivity, ability to handle dynamic objects, and capability to process a large number of 3D CAD model shapes, sizes, and forms have been identified as essential technical requirements of any method designed for collision detection and interference analysis of 3D virtual construction simulations [26]. Also imperative is the ability to perform proximity queries in real-time by processing position and orientation data. Output from the analysis stage such as proximity between entities, potential collision, and breaching of safety thresholds can be used by downstream processes such as operator warning frameworks and equipment control mechanisms. The following sections describe the technical details of individual components of the proposed approach.

5.1 Track

Excavation operations typically involve infrastructure that is covered by earth, mud, soil, and debris. Equipment is used to remove the overlying cover. Excavation equipment has an end effector (e.g., bucket) that interacts with the materials overlying the buried infrastructure. Operators of such equipment cannot be certain of the exact location of the end effector in relation to buried infrastructure. In order for such operations to be carried out in a safe manner, accurate knowledge of the pose (position and orientation) of the equipment's end effector is critical. Knowledge of the end effector's pose can then be used in the analysis stage to compute spatial relationships with infrastructure in its immediate surroundings. Hence, the primary step in carrying out geometric computations between job-site entities is tracking the poses of equipment and in particular their end effectors.

5.1.1 *End effector position and orientation*

Due to the harsh nature of the work environment and operation, sensors cannot be typically placed directly on the digging implement of an excavator. To arrive at the pose of the end effector, a combination of multiple sensors and real-time computations is employed. An excavator's end effector is its bucket (bucket teeth). However, it is not practical to place GPS sensors in the bucket owing to the potential damage due to constant interaction with the soil. The end effector's pose is obtained through use of a GPS sensor placed on the excavator cab and orientation or tilt measuring sensors placed along the rotating joints for the boom, stick, and bucket as described in Talmaki and Kamat [53]. The use of an indoor GPS tracking system has been demonstrated to calibrate the sensors on-board a backhoe loader and track its position and orientations while performing operations in real-time [28].

As the pose of the end effector cannot be directly obtained, computations are made with output from multiple position and orientation sensors placed on the equipment as schematically represented in Fig. 3. Real-time calculations are performed to derive the end effector pose. Thus, the tracking stage is responsible for determining the pose of the dynamic entity's end effector or component that interacts with other job-site entities. Pose determination is done through a process of forward kinematics.

In forward kinematics, the final shape of the articulated chain is dependent on the angles explicitly specified for each of its links [25]. The output has two components to it—position and orientation. A six-parameter vector is used to describe the spatial location and orientation of an object in space. A similar approach has been successfully implemented in autonomous navigation and aircraft autopilot systems [13]. The six parameters making up the vector are latitude, longitude, altitude, roll, pitch, and yaw.

Position sensors such as GPS compute locations in terms of geodetic latitude, longitude, and altitude above an imaginary ellipsoid. These locations are provided with respect to a specific geodetic datum or reference coordinate system. For GPS, the datum is WGS-84 [12]. Position coordinates are sometimes converted from spherical units such as latitude–longitude to planar systems such as universal transverse mercator (UTM), which are more accurate at representing smaller areas by preserving angles and not distorting shapes. Planar systems employ linear units such as meter and foot and thus measurements made in them can be directly used in distance and other computations unlike spherical units [17].

Orientation is defined in terms of a triplet of parameters that are traditionally used to illustrate rotation of an aircraft in aeronautical domains [46]. The triplet of roll, pitch, and yaw compute the rotation of a body about its longitudinal, transverse (lateral), and vertical axis, respectively. Thus, the output from the 'Track' stage is used to update dynamic entities and analyze their interactions with other job-site entities, and also to create a graphical simulate of the job site.

5.2 Representation in 3D

The second stage of the proposed technical approach is representing the job-site operation being performed in a real-time 3D visualization. The use of telepresence and virtual reality has been shown to have promise for visualization and study of hostile and extreme environments [20]. Virtual reality allows users to see beyond what is possible through conventional video camera feeds. The authors envision that a virtual reality scene representing a city block can be manipulated by a utility inspector to alter the transparency of the surface of terrain and to show buried infrastructure in that

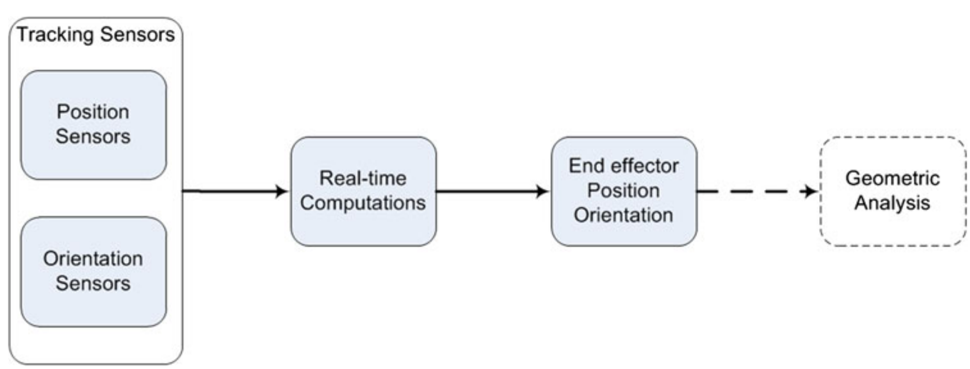

Fig. 3 End effector position–orientation computation through sensor fusion and real-time computation

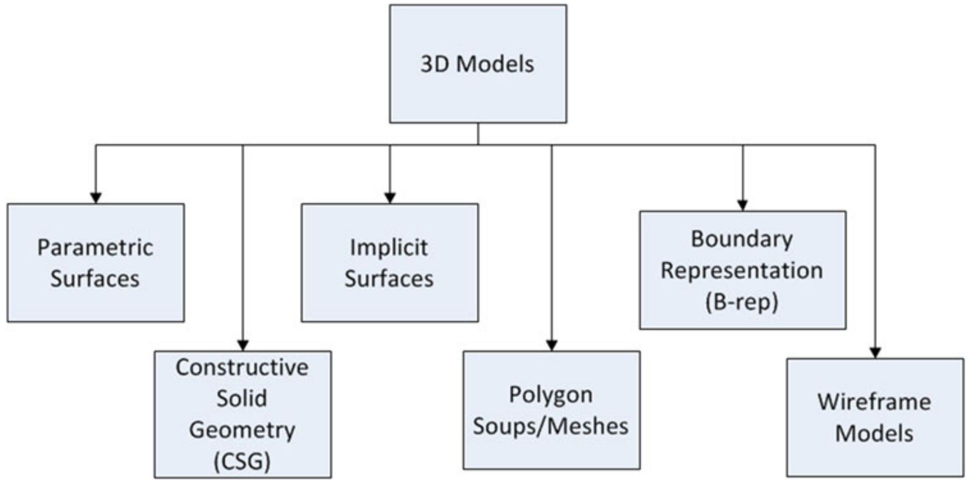

Fig. 4 Classification of 3D model representations

area. 3D virtual scenes also enable operators to enjoy views of their equipment and surroundings that are not feasible through placement of conventional cameras that have a much narrower field of view [23].

5.2.1 Types of 3D models

3D models or geometry can be represented in a variety of formats such as wireframe models, surface models, solid models, meshes, and polygon soups. Depending on the level of realism required, 3D models can vary in accuracy, memory requirements, difficulty of modeling, and use in real-time rendering and analysis. However, the choice of 3D geometry format also depends on the end application and real-world entity being modeled. Polygonal meshes are the most commonly used type of format to represent 3D geometry. Their popularity is also enhanced due to their versatility and ease of rendering [32, 37, 49].

Parametric representations of surfaces typically define a surface as a set of points and are commonly used in many commercial modeling systems [3]. Non-uniform rational B-splines (or NURBS) is an example of parametric methods. Applications requiring the use of a truly smooth surface at every scale find NURBS to be a convenient option to implement [37]. Implicit surfaces define a set of points that satisfy a function F where $F(x, y, z) = 0$ and all points that satisfy the criterion $F(x, y, i) < 0$ define a solid that is bounded by the implicit surface. Constructive solid geometry (CSG) allows 3D shapes to be built from simpler primitives such as cubes, cylinders, and spheres through the application of boolean operations on them. 3D file formats used within the CAD domain often store their geometric data such that CSG principles can be used to operate upon them [37].

Boundary representation (or B-rep) describes a solid in terms of its surface boundaries, i.e., vertices, edges, and faces [14]. In B-rep a data structure containing information about an object's edges, faces, vertices, and their topological relationships is used to represent it. Figure 4 shows the classification of 3D models into the above-described formats. For a detailed review of 3D model types, the reader is referred to Foley et al. [16] and Hearn and Baker [19]. Polygon models are the most suitable format for hardware-accelerated rendering and the most commonly used format for representing 3D models [32, 37, 49]. Hence, the polygon soups class of polygon surface models were selected as the 3D model format to represent real-world entities and perform geometric analysis.

5.2.2 Georeferenced 3D models: static entities

Dynamic entities on a job site are tracked using position and orientation sensors. Their equivalent 3D models in a real-time simulation are updated through position and orientation output from the 'Track' stage of the framework. However, to monitor spatial relationships between static and dynamic entities, equivalent positional information must be made available regarding the location of static entities. This is made possible through the use of georeferenced or location-aware 3D models. Georeferencing refers to the ability to assign a location attribute to data. The location attribute can be obtained through archived data sources such as geographic information system (GIS) databases, infrastructure repositories, and as-built drawings. Static entities such as underground utilities, embedded rebar, temporary structures, and material stock piles have their location information linked to the earth's surface.

Thus, georeferencing ensures that 3D models representing static entities can be used in the same analysis with dynamic entities that utilize position–orientation information having a common coordinate system. Specifying the location of an object on the earth's surface is done through coordinate systems such as geographic coordinate systems

(GCS) that use spherical units (latitude–longitude) or projection coordinate systems (PCS) that use linear units. When creating a 3D simulation, it is important to note that all position–orientation data and georeferenced 3D models are sharing the same coordinate system and units. Dissimilar systems and/or units can result in 3D models appearing further away than in reality and the resulting analysis will also produce unusable output. Thus, spatial analysis between dynamic and static entities is made possible through the use of georeferenced 3D models.

5.2.3 3D model updates: dynamic entities

In order for a 3D visualization to simulate an ongoing operation, all dynamic entities must update their pose to match their real-world counterparts. In addition to the 3D visualization stage, updates made to dynamic entities are used downstream in the 'Analysis' stage. 3D models representing dynamic entities thus afford equipment operators a virtual 3D view of the operation they are performing and concurrent geometric proximity analysis between job-site entities.

5.3 Analyze

The analysis of interference and/or interaction between geometric entities is intended to help equipment operators by providing improved spatial awareness. This can only be achieved if all computations are carried out in real time (i.e., concurrently with the real-world operation). It thus follows that efficiency of the algorithms implemented in the analysis stage is critical. At the same time, accuracy of the chosen algorithm cannot be sacrificed to achieve optimum real-time or near real-time performance.

5.3.1 Creation of 3D graphical database

A class of polygon surface models known as polygon soups is made up of several hundred geometric primitives (e.g., triangles and quads) that are often present in an un-orderly arrangement with no topological relationships between adjacent polygons. Figure 5 shows the 3D model of a dump truck and its corresponding polygon surface model showing the composition of its surface. Such a collection of polygons is commonly referred to as a polygon soup. When a proximity query is made between a pair of entities, the objects' geometric primitives and their global position and orientation combine to create the graphical database for the query at hand as shown in Fig. 6.

Computational queries provide valid output as long as their underlying graphical database is current in terms of geometric makeup and object pose. The geometric primitive component of the graphical database remains the same unless there is a change in equipment type or configuration (e.g., a change of end effector for an excavator). The pose is updated in real-time and maintains the fidelity of the database. Thus, it is important to point out that both geometric content and pose information must represent their real-world counterparts at all times to ensure meaningful analysis. Figure 6 shows the breakdown of 3D models into their geometric primitives for illustration purposes only as the analysis of the graphical database is performed in a multi-stage manner and involves operations at the object and sub-object level, and not always at the primitive level as seen in the following section.

5.3.2 Analysis of 3D graphical database

Algorithms for the detection of interference between objects are an essential component in virtual environments. Their applications are wide spread and can be found in areas ranging from surgery simulation and games to cloth simulation and virtual prototyping [58]. The set of spatial queries between a pair of objects includes collision detection, exact separation distance computation, and approximate distance measurement to a tolerance value [30].

A bounding volume (BV) is used to bound or contain sets of geometric primitives such as triangles, polygons,

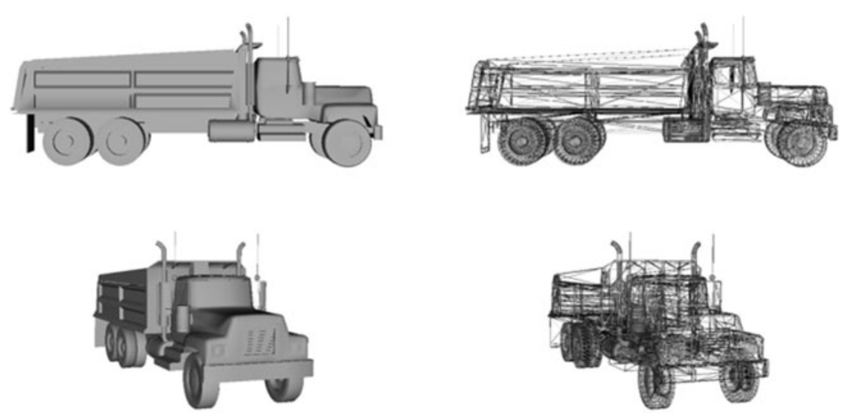

Fig. 5 Polygon surface models and their corresponding polygonal/wireframe view

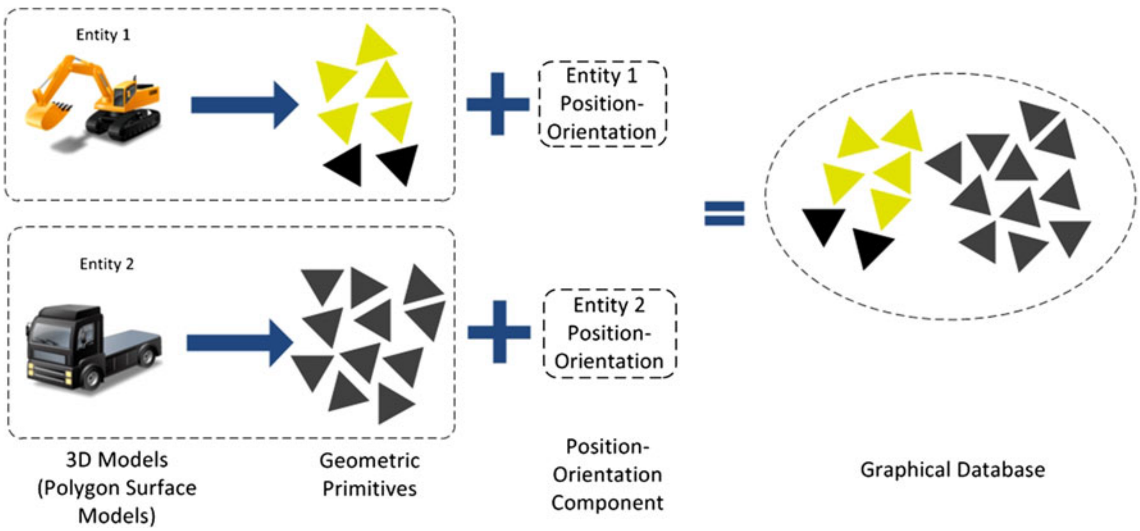

Fig. 6 Creation of graphical database from 3D models' polygon soups and their global position–orientation. The colors given to the triangular primitives are intended to represent the surface color of the entities they represent—*yellow* + *black* for entity 1 and *gray* for entity 2 (color figure online)

and NURBS [30]. The most common types of bounding volumes (BVs) are spheres, axis-aligned bounding boxes (AABBs), oriented bounding boxes (OBBs), discrete-oriented polytopes (k-DOPs), ellipsoids, convex hulls, and sphere swept volumes (SSVs) [30, 58]. The efficiency of a bounding volume is affected by the choice of a BV type. Efficiency is achieved by a trade-off between tightness of fit and speed of operations between two such BVs. No single BV is optimal for all situations and queries [30].

The underlying algorithm in the developed methodology uses SSVs as the BV type. First- and second-order statistical methods are used to compute the BVs. The mean and covariance matrix are computed for vertices of all triangles making up the object and used to summarize the object's primitive vertex coordinates. Computation details and formulae are described in detail in Gottschalk et al. [18] and Larsen et al. [30]. An OBB is first constructed to enclose the underlying geometry using the eigenvector of the covariance matrix computed for vertex coordinates. The second step consists of creating BVs of SSV type.

There are three types of SSV: point, line, and rectangle SSV. Selection of SSV types is based on the dimensions of the OBB constructed. The BVs are arranged in a hierarchical manner beginning with the root node that encompasses all geometry associated with an object and ending with leaf nodes that contain only a single triangle. The BVs thus created at different node levels form a bounding volume hierarchy (BVH) using a top-down strategy. All triangles in a given node are split into two subsets where each subset then becomes a child node of the current node. This subdivision continues until a given node contains only a single triangle or primitive and can no longer be further subdivided.

Triangles in a node are subdivided based on a splitting rule described in Gottschalk et al. [18]. The first step of the subdivision process is the selection of a suitable axis. The subdivision rule that is adopted in this procedure uses the longest axis of a box (i.e., OBB created for the geometry). If the longest axis cannot be selected then the second longest axis is chosen, else the shortest axis is chosen. In the next step, a plane orthogonal to the selected axis is constructed. This plane acts as the partitioning plane such that polygons are divided into two groups according to which side of the plane their center point lies on. The subdivision process continues until a group of polygons cannot be further partitioned by this criterion (i.e., the group is considered indivisible or it encounters a leaf node that contains only a single triangle/primitive).

Traversal of a BVH during a proximity query is referred to as bounding volume tree traversal (BVTT). The traversal begins at the root node and proceeds along the tree in a depth-first or breadth-first manner. The distance query returns the exact distance between a pair of objects, while the collision query returns whether they are overlapping or not as illustrated in Fig. 7.

Overlap and distance computations are performed using a multi-stage approach. The first stage is less computationally intensive and is used to eliminate a large number of test pairs from being passed on to the more computationally demanding second stage. In the first stage, OBBs are used to determine if two objects are disjoint or overlapping. The same computation is used by overlap as well as distance tests to eliminate or prune object pairs from further checks. The algorithm is based on the separating axis theorem that is described in detail in [18]. A pair of bounding boxes to be checked for overlap is projected onto

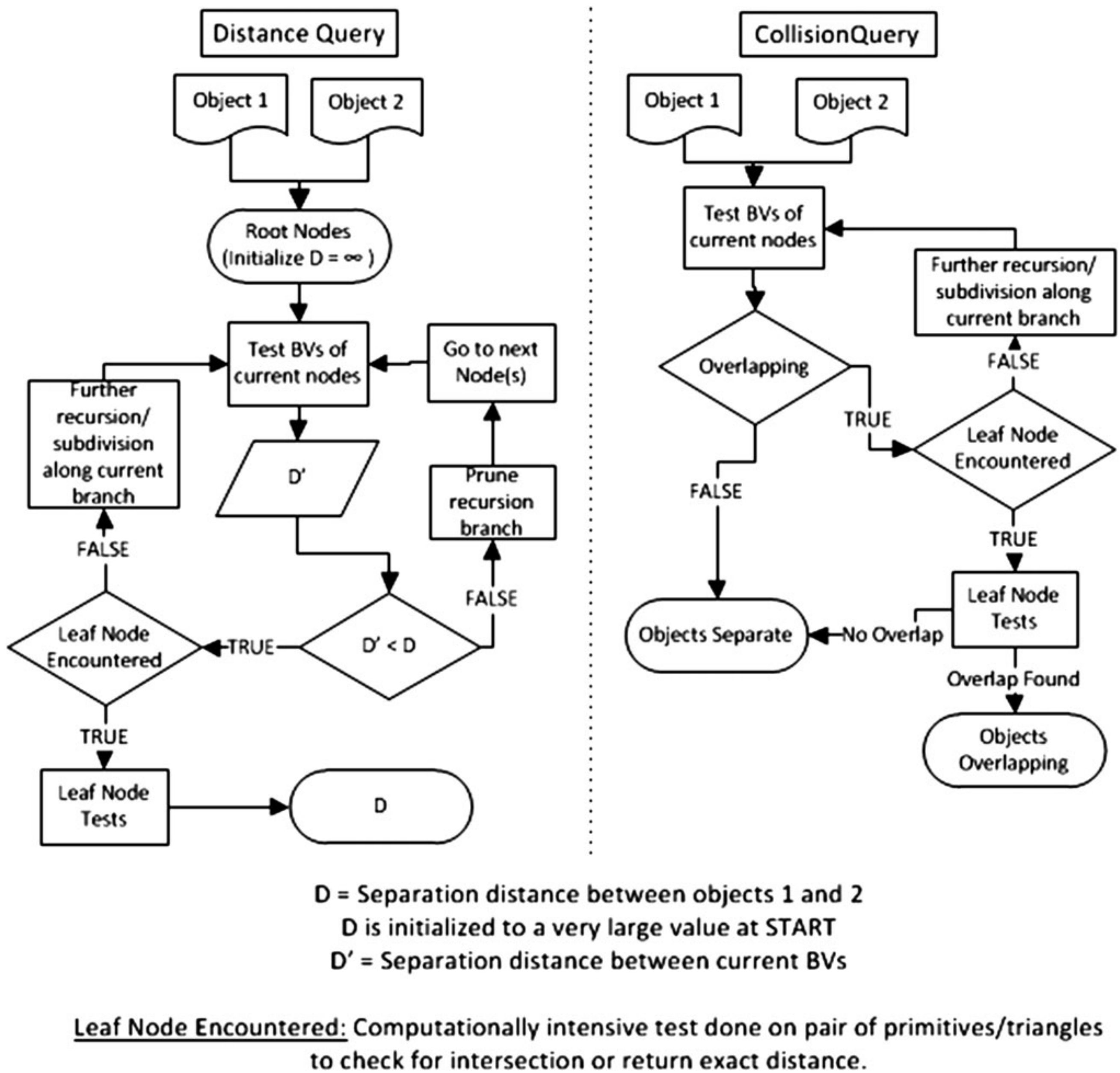

Fig. 7 Flow chart illustrating the progression of distance and collision queries in underlying algorithms

a random axis in 3D space. In this axial projection, each box forms an 'interval' on the random axis. An interval is the distance between two projected points on a random axis. If the intervals of the boxes do not overlap then the pair is disjoint/separate for that given axis. Such an axis is termed as a 'separating axis'. The separating axis procedure is graphically represented in Fig. 8. Once an overlapping axis is found, no further tests are required to check for overlapping.

If on the other hand, the intervals are found to be overlapping, then further tests are required to determine concretely whether the pair is separate or overlapping. Two boxes are disjoint if and only if a separation axis exists which is orthogonal to a face of either box or orthogonal to an edge from each box. Each box has three unique faces and three exclusive edge orientations. Thus, a total of 15 (six face combinations and nine pair-wise edge combinations) separating axes exist for a given pair of boxes. If no separating axis exists, then the boxes are overlapping and if the current OBB is a leaf node (i.e., bottom-most node enclosing a geometric primitive), then the pair is passed to stage two for further computations.

The second stage is more computationally demanding and checks for overlap between underlying geometric

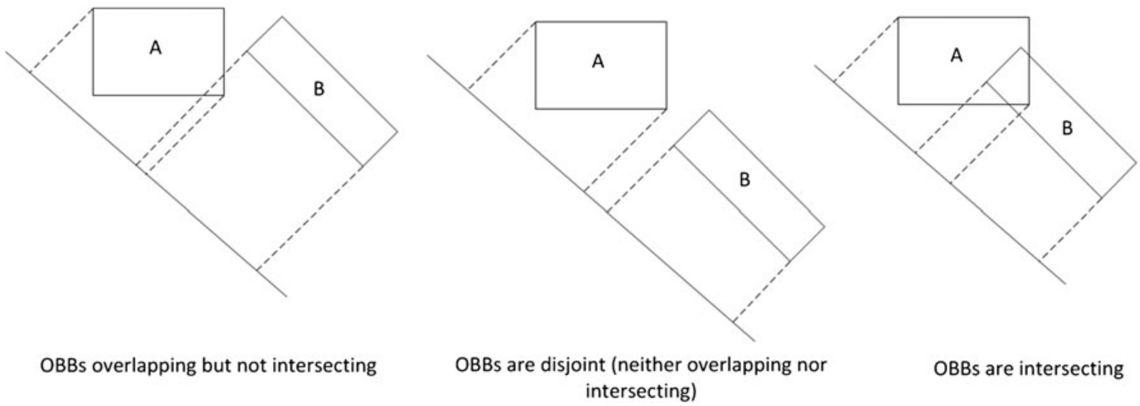

Fig. 8 Use of separating axis to determine overlap between OBBs

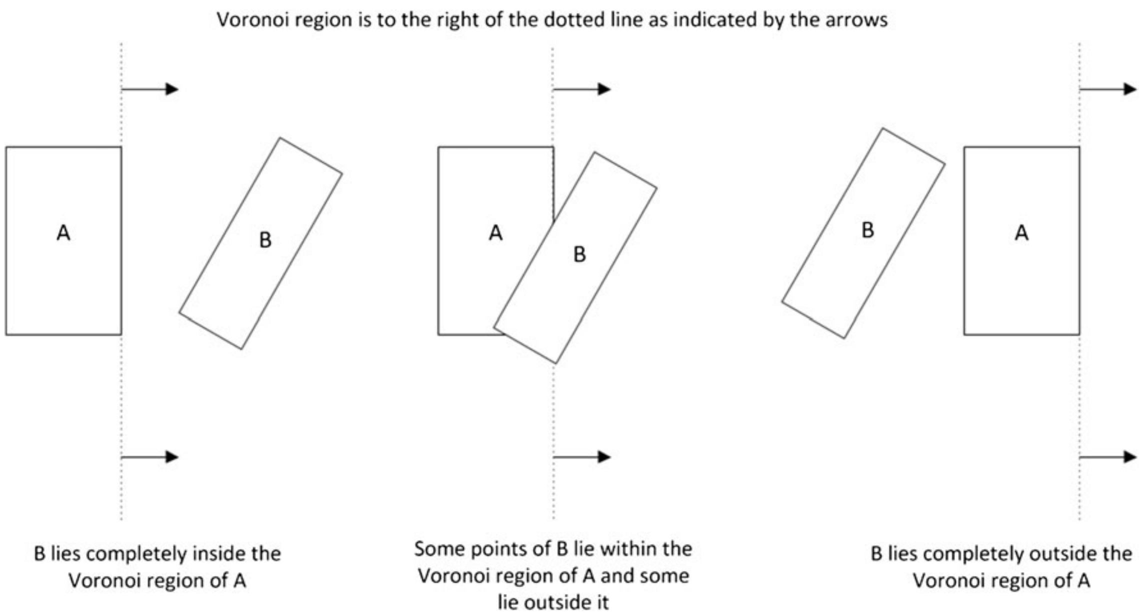

Fig. 9 Use of external Voronoi regions to determine configuration of pairs of edges

primitives. If primitives are found to intersect, i.e., no separation distance between primitives, the computation, which uses a list of primitives making up the object, determines which primitives on each object are intersecting and their exact points of intersection. If primitives are found to be separate, the separation distance between them is calculated. Computation and algorithm details involved in the stage-two leaf-node tests are described in greater detail in Larsen et al. [30]. The algorithm first determines if the closest points between the primitives lie on their boundary edges. This is determined through the use of external Voronoi regions. An external Voronoi region for an edge A of a primitive is defined as the region of space outside the primitive in which all points are closer to edge A than any other features of the primitive.

According to the lemma used in the algorithm in Larsen et al. [30], there are three possible configurations for a pair of edges A and B belonging to two primitives—(1) edge B is present entirely inside the Voronoi of A; (2) some points of B are inside and some outside the Voronoi of A; and (3) B is present entirely outside the Voronoi of A as shown in Fig. 9. From a possible 16 pairs of edges, if no pairs satisfy the requirements of the lemma, then primitives are either overlapping or the closest point lies in their interior. Cases one and two are simple acceptance rejection configurations while case three requires additional tests to be performed.

To determine if primitives are overlapping or separate, a modification of the separating axis theorem used in stage one for OBB comparison [18] is used. Primitives are projected to a unit direction and the distance between intervals is computed. This value represents the lower bound for the actual separation distance [30]. If one of the closest points is in the interior of a primitive, the maximum magnitude of the normal vector to each primitive gives the upper bound of the separation distance. If both the lower and upper bound values are zero, then the primitives are overlapping.

The algorithm described in Larsen et al. [30] uses optimization techniques to speed up computations and by pruning nodes sooner in the tree traversal. It optimizes proximity queries through a technique called priority directed search. Priority queue is a variation of the conventional queue data structure in which elements have a user-determined priority associated with them. Elements with higher priority are given preference over elements with lower priority. The priority in this case decides the order in which proximity tests are to be performed. The algorithm for proximity queries assigns priority based on distance from the current BV pair. Closer BV pairs are given higher priority and checked prior to BVs lying further away.

Another technique used to optimize proximity queries takes advantage of coherence between successive frames of motion. The distance between a pair of objects changes by relatively small amounts between successive frames due to the high frequency of performing proximity queries. The closest pair of triangles from a given query is recorded or cached and their corresponding separating distance is used to initialize '*D*' rather than initialization with a very large value as shown in Fig. 7. This technique is known as triangle caching. As the distance moved is very small, the value of '*D*' from a prior frame is very close to the current '*D*' between a pair of objects. Details of the priority-directed search and triangle caching optimizations can be found in Larsen et al. [30].

6 Protocol

Figure 10 shows a real-time simulation framework to assist excavation operators carry out operations in the presence of buried utilities, avoiding unintended strikes [52]. The framework is generic and can be used to simulate any construction operation where operators experience reduced visual guidance and spatial awareness. SeePlusPlus is the 3D visualization component of the framework that allows users a virtual, interactive view of the ongoing operation. S2A2 (sensor stream acquisition allocation) Framework is responsible for transmitting position and orientation sensor data from the real-world job site into the virtual world. Thus, input from S2A2 is used to update 3D models in the visualization and real-time spatial queries.

There are dedicated modules for creation of static and dynamic 3D entities. The B3M Toolkit is used to create georeferenced 3D models of buried utilities. These models can be directly used in the simulation as they are location-aware. 3D models that are location aware can be loaded into a scene without the need to be manually given a position and/or orientation. Their location in the real world

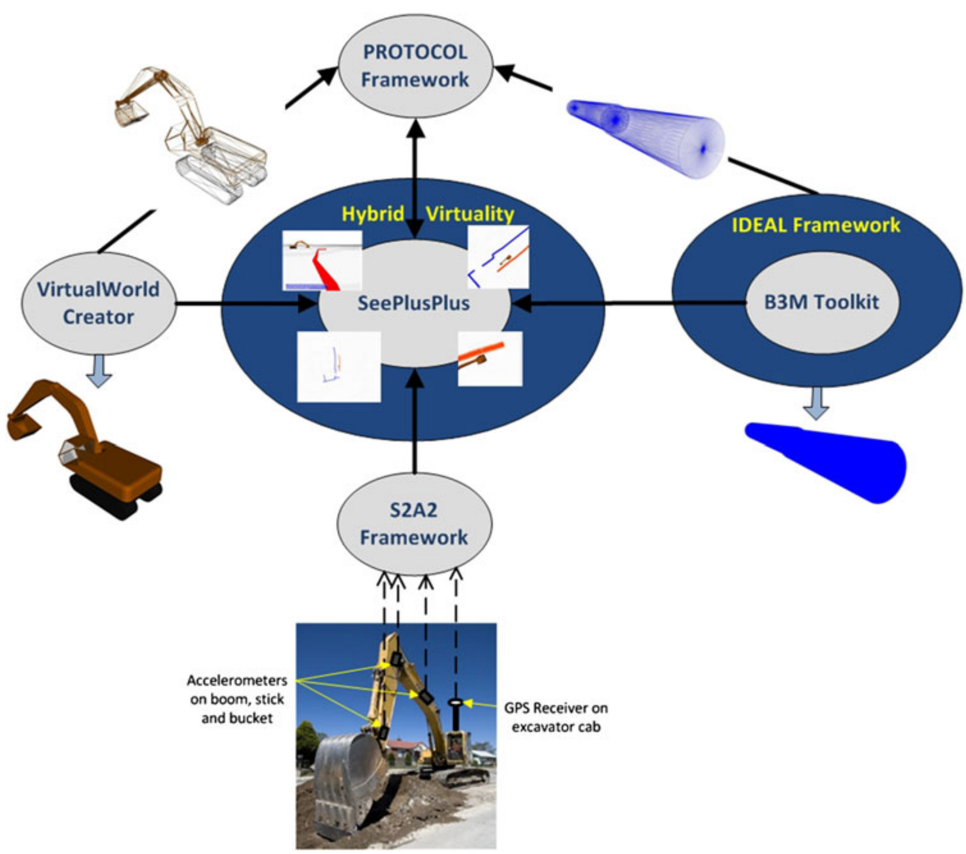

Fig. 10 Constituent modules making up the overall framework to assist excavation operators in avoiding buried utility strikes

is represented in the 3D models when they are created. VirtualWorld Creator provides 3D models representing articulated construction equipment used in the operation. While Fig. 10 shows an excavator being introduced, it can provide the appropriate 3D equipment model based on the type of operation being simulated. PROTOCOL is the module responsible for providing spatial queries using 3D models representing static and dynamic entities, receiving sensor input from the real world, and providing audio-visual warning feedback to the user for accident avoidance.

Thus PROTOCOL is a software module that is designed to integrate with a 3D visualization system and allow users to make spatial queries. Users can select between proximity monitoring, collision detection, and tolerance checking queries. The PROTOCOL name was chosen to reflect the three categories of queries available. PROTOCOL's geometric computational functionality is designed using the Proximity Query Package (PQP) library [42]. The PQP library uses geometric primitives making up an object and its global pose to compute proximity between two or more objects.

As described in Sect. 5, the use of low-level geometric primitives for distance computation and collision detection ensures that these computations can be made to a much higher resolution than that possible through use of bounding volumes alone. However, the use of polygon surface models to represent static and dynamic entities means that 3D objects need to be decomposed into their primitives to be used by the algorithms. The decompositions represent the polygon surface model as a list of its constituent triangle primitives and are stored in memory as 'tris' files. A 3D model composed of quads or other non-triangular primitives is also automatically decomposed into

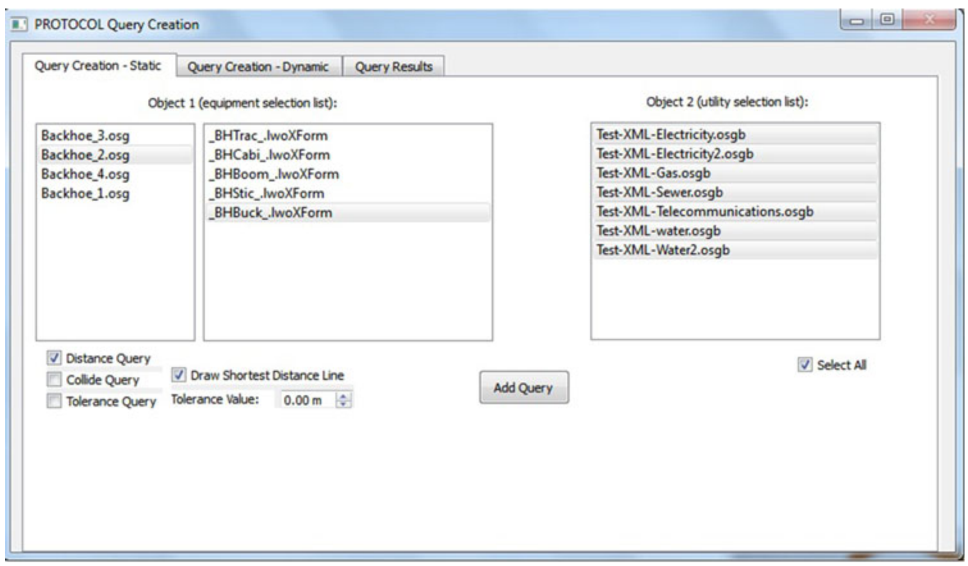

Fig. 11 PROTOCOL graphical user interface for spatial query instantiation between dynamic and static entities

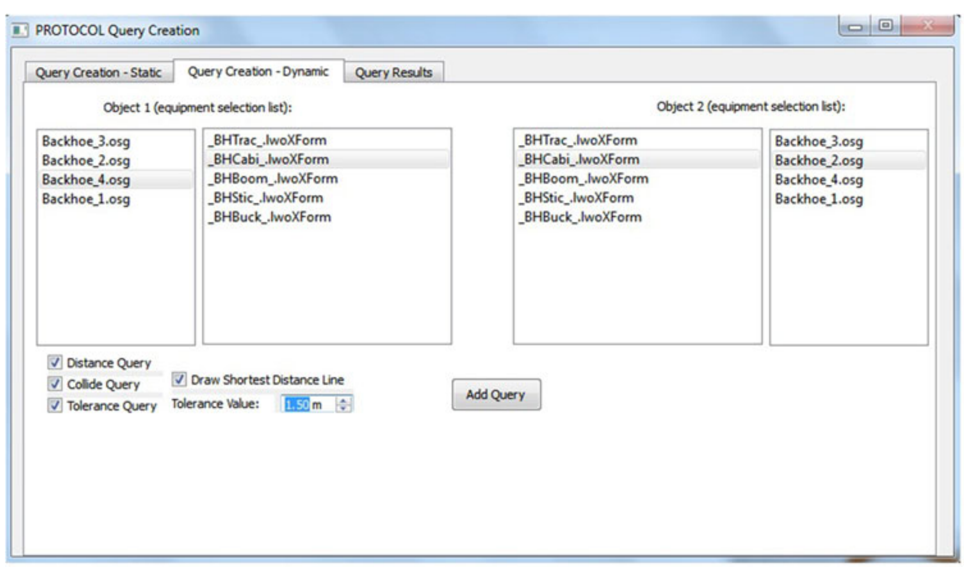

Fig. 12 PROTOCOL graphical user interface for spatial query instantiation between dynamic entity pairs

its equivalent list of triangles. The geometric primitives representing a rigid body remain consistent throughout its operation. Thus, pose updates from real-world sensors when combined with tris files can represent a real-world entity's pose in a virtual world. Every tris file can be used for multiple instances of the same 3D model by simply substituting each instance's position–orientation.

PROTOCOL is implemented as a plugin to SeePlusPlus and is presented to the user as a graphical user interface (GUI) where queries can be created between entities. Figures 11 and 12 show the PROTOCOL GUI for static query and dynamic query creation, respectively. One-one and one-many relationship queries can be created. Any combination of distance, collision, and/or tolerance queries can be instantiated between two entities. Tolerance queries can be used to check if a predetermined safety threshold distance has been breached during operation.

Figure 13 shows the list of queries created and entities participating in a given query. The GUI provides an option to render a line joining the closest points between a pair of objects that is aimed at assisting operators in identifying the entities that are participating in the query when the number of entities is high. In addition, queries created in PROTOCOL can be saved and re-used for future use. This feature is envisaged as being useful in scenarios where the end user such as an equipment operator can load a pre-created virtual scene and proximity queries without having related expertise in those areas.

7 Validation experiments

The two key parameters to measure a proximity monitoring framework's effectiveness are measurement error and latency. Measurement error in this case refers to the error of the computed distance with regard to ground truth or theoretically expected values. Latency in this case is a measure of time lag between an event occurring in the real world and a proximity monitoring framework providing output to warning systems that end users depend upon. This section describes experiments carried out to test the measurement error and latency of the PROTOCOL module. The experiments were carried out at the Intelligent Sensing and Automation Testbed (ISAT) at the National Institute of Standards and Technology (NIST) in Gaithersburg, Maryland. The following sections describe the details of the experimental setup, methodology, and results.

7.1 Proximity test

PROTOCOL's proximity measurement performance is evaluated in this experiment. The metric for PROTOCOL's proximity measurement performance is the difference between PROTOCOL's reported proximity and ground truth or theoretically expected values. Any variations between computed and expected values will demonstrate PROTOCOL's contribution to computational errors. However, to ensure that any errors are purely computational and not originating from position and orientation tracking, a tracking system with sufficient accuracy must be used. Although modern, commercial GPS units are capable of sub-centimeter position accuracies using certain technologies such as differential-GPS (DGPS) and real-time kinematic GPS (RTK-GPS), their accuracies can be reduced due to environmental factors, surroundings, and line-of-sight to the sky [33]. Thus, the use of outdoor GPS or technologies that cannot guarantee a consistently high level of positional accuracy can adversely affect the

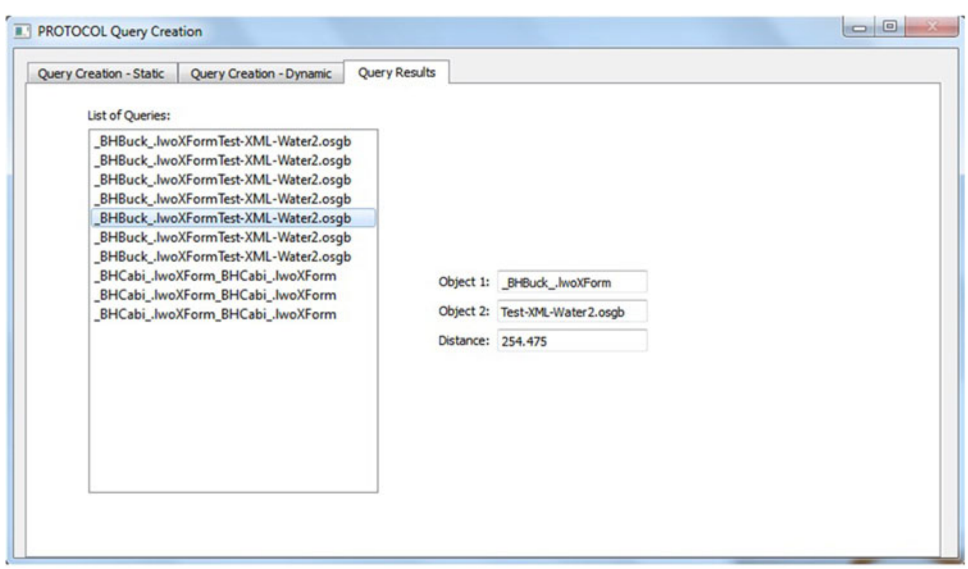

Fig. 13 PROTOCOL graphical user interface for displaying list of queries created and entities involved in the query

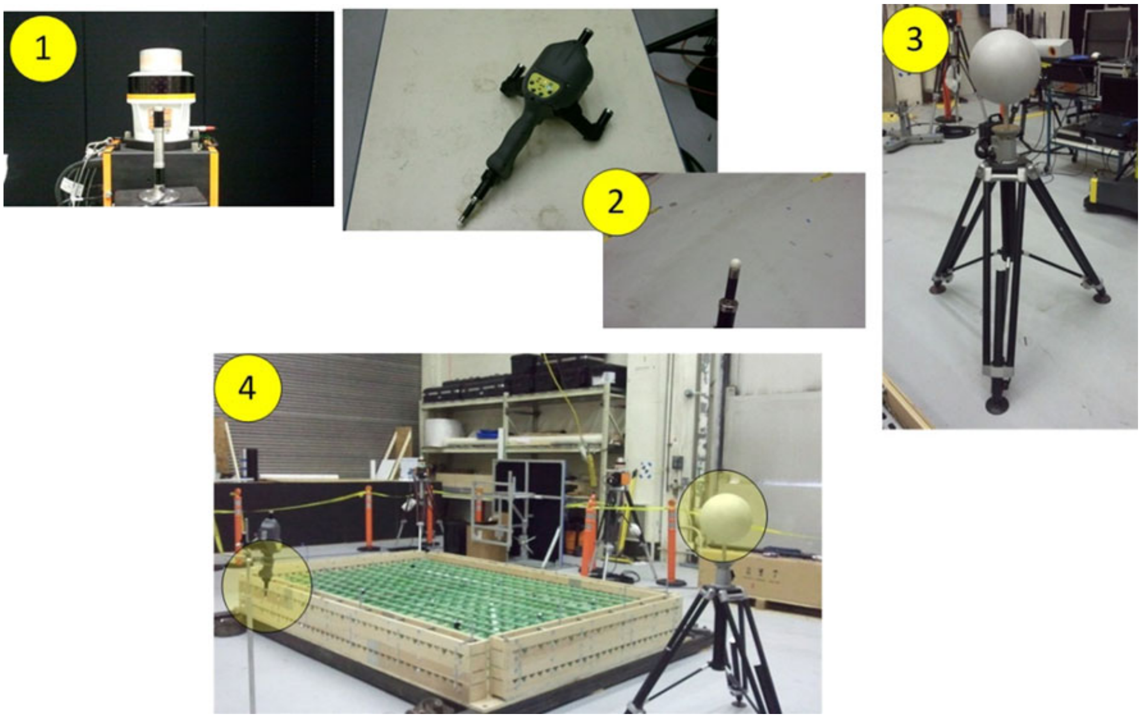

Fig. 14 *1* Indoor GPS transmitter; *2* indoor GPS tracking probe (i6 Probe) and its tracking tip; *3* test sphere mounted on tripod; *4* experiment setup with i6 Probe placed adjacent to test sphere

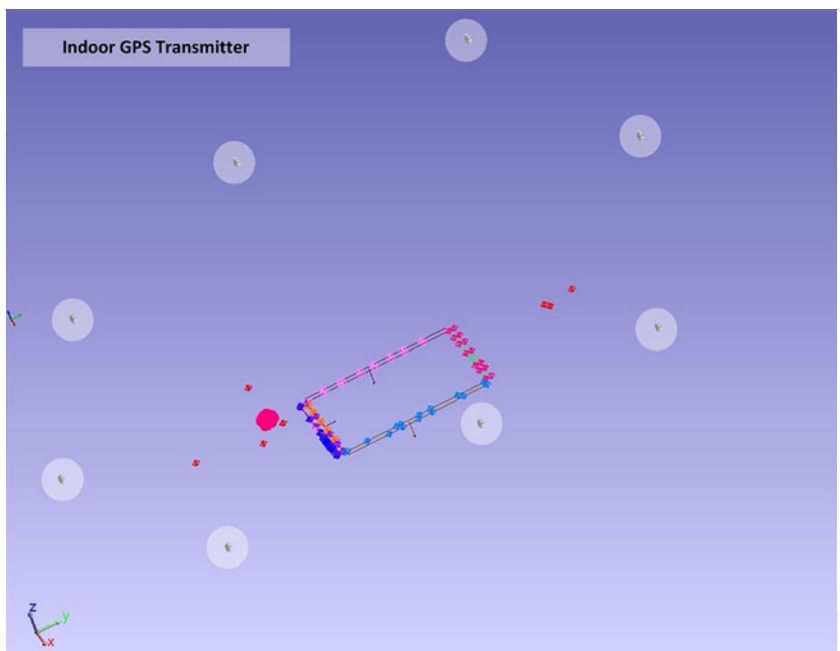

Fig. 15 Graphical representation of the layout of the eight iGPS transmitters (*large circles*) in the ISAT with some of the measured data shown as *dots* in the figure

computations performed by PROTOCOL. If position tracking is inaccurate and the magnitude of uncertainty cannot be measured, the difference between PROTOCOL computed distances and theoretically expected values cannot be attributed to either positional input or PROTOCOL computations.

To ascertain the cause of uncertainty, a tracking system capable of consistent readings was required for this experiment. Indoor GPS (iGPS) is one such tracking technology that is capable of sub-millimeter ($\pm$0.250 mm) position tracking accuracy [41]. iGPS can be compared to the GPS constellation of satellites where

Table 4 Comparison of PROTOCOL computed values and ground truth values for various separation distances

Iteration #	Nominal separation distance (m) (ft)	Ground truth (average for ten readings) (mm) ± 0.354 mm	PROTOCOL computed (average for ten readings) distance (mm)	Δ (mm) (computed—ground truth)
1	<0.305 (1)	290.360	292.420	1.560
2	0.305–0.914 (1–3)	724.730	725.878	1.148
3	0.914–1.524 (3–5)	1,188.768	1,189.552	0.784
4	1.524–2.134 (5–7)	1,856.836	1,857.434	0.598
5	2.134–2.743 (7–9)	2,475.590	2,476.090	0.500
6	2.743–3.353 (9–11)	3,032.077	3,032.490	0.413
7	>6.096 (20)	7,073.196	7,073.520	0.324

The standard deviations for averages in columns 3 and 4 range from zero to negligibly small and are thus not shown in the table

each indoor transmitter plays a similar role to a GPS satellite. Instead of satellites, iGPS uses infrared laser transmitters that emit laser pulses. When a receiver obtains pulses from more than a single transmitter, it can compute its position and orientation within the coordinate system defined by the transmitters. Photo detectors pick up the signals and compute angle and positions based on the timing of the arriving light pulses. Figure 14 shows the instruments and sensors used to conduct the experiment. Figure 15 is a graphical, overhead view of the ISAT testbed.

The radius and center of the test sphere in Fig. 14(3) were measured using the iGPS. Distances measured with the iGPS were treated as ground truth. The test sphere and probe tip were modeled as polygon surface spheres to be used in geometric proximity analysis and real-time 3D visualization. The proximity test consisted of a series of seven iterations starting at a distance of <0.3048 m (1 ft) surface-to-surface separation between the test sphere and the probe tip. Successive iterations were made at 0.6096 m (2 ft) increments. For each iteration, ten readings were made to account for variations in the iGPS tracking. For every measurement, the error which is the difference between the value computed by PROTOCOL and the ground truth was calculated. Table 4 shows the separation distance error for the complete range of separation distances. Since the iGPS positional uncertainty for a given point is ±0.250 mm, the difference between two iGPS points can vary by a combined ±0.354 mm, where $0.354 = \sqrt{(2 \times (\pm 0.25^2))}$.

7.2 Latency test

A good proximity monitoring system must be capable of analyzing input and providing real-time or near real-time output in order for it to be useful. Bryson [5] identified the two main causes of dynamic distortion in real-time computer graphics system as lag and frame rate. In these experiments, lag was defined as the time offset in the data stream from a tracker to the graphical image. It was also observed that frame rate depended upon the scene complexity, varying from very simple rendering to those requiring considerable computation and rendering in response to user movement [5].

In its basic form, latency is a measure of overall delay in a system. However, its constituent time delays and their measurement are system and implementation specific. In the case of PROTOCOL, the delay is measured between an event occurring in the real world and a warning being given to the user by the system. For example, if PROTOCOL is set to a tolerance mode of 1 m, the system must provide output for audio/visual warnings when two entities are within 1 m or less of each other. The time delay between entities coming within the tolerance distances in the real world and audio/visual warnings being provided to the user is an example of overall system latency as represented in Fig. 16.

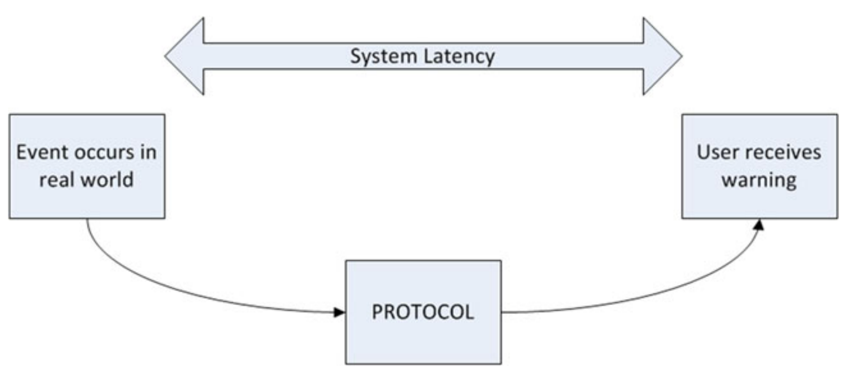

Fig. 16 Time lag between an event occurring in the real world (e.g., job site) and the user receiving a warning is defined as system latency

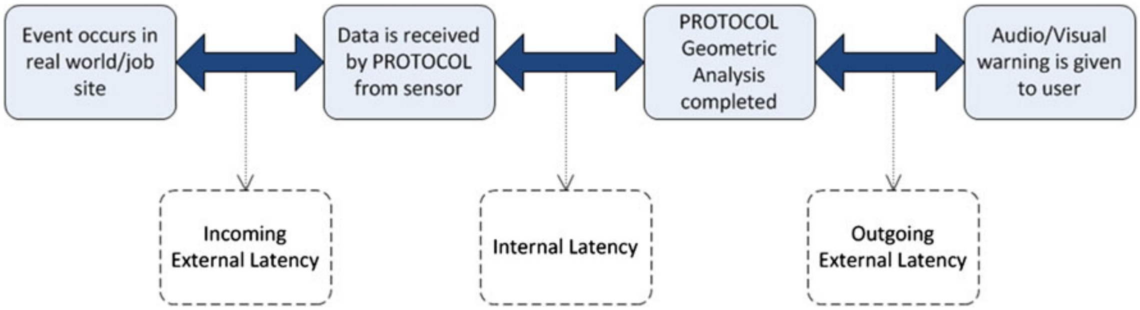

Fig. 17 Breakdown of system latency into constituent components

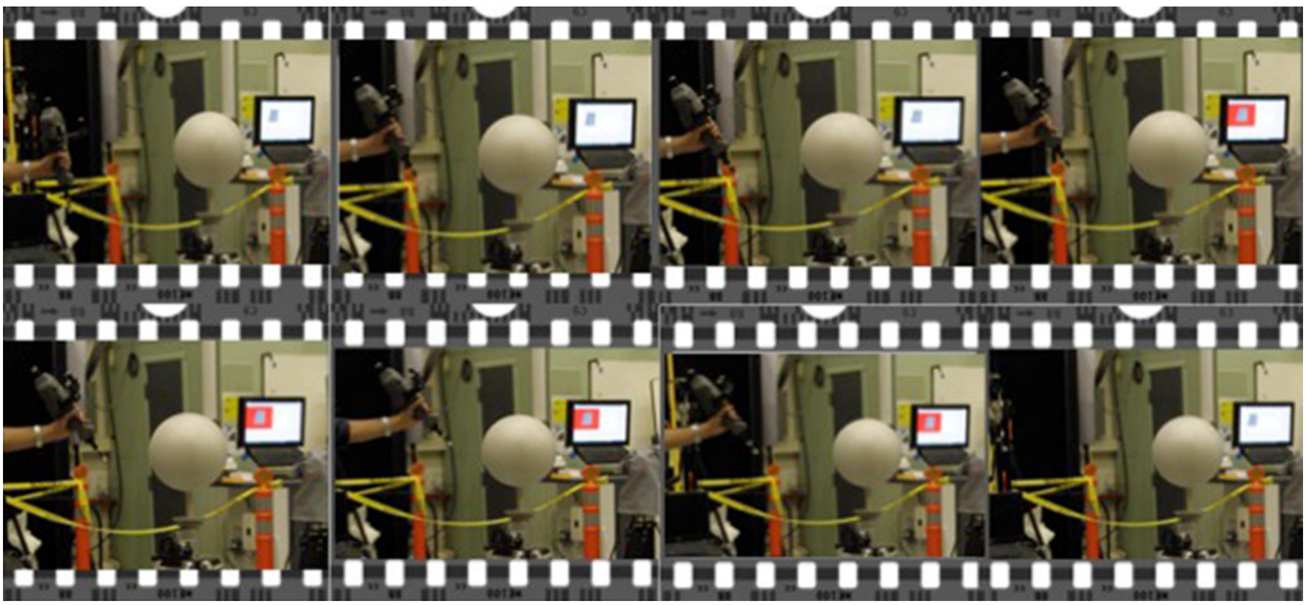

Fig. 18 Image captured from a video recording of the latency test showing change in background color from *white* to *red* signifying breach of preset safety threshold (color figure online)

System latency in relation to PROTOCOL is further sub-divided into incoming external latency, internal latency, and outgoing external latency based upon the time taken to complete specific tasks as shown in Fig. 17. Incoming external latency refers to the time taken for sensors placed on or around equipment to register a change in position–orientation and transmit it to PROTOCOL. It is a function of the data transmission medium chosen and the refresh rates of the sensors. Internal latency is the time taken by PROTOCOL to update the existing graphical database and perform geometric analysis. Hence, it is a function of scene complexity and number of entities being modeled, and the number and types of queries being performed. Finally, outgoing external latency is the time taken for transmission of analysis output to audio/visual systems that process and supply appropriate warnings to the user. It follows that it is a function of data transmission medium and types of warning devices being implemented. A visual warning mechanism that is part of the visualization framework can reduce the time to negligible values while disparate warning devices such as audible alarms can add additional time due to additional transmission being involved.

In this test, a tolerance trigger was used to measure system latency. The experiment setup was identical to the distance test, including the i6 Probe, test sphere, and iGPS transmitter layout. The test was designed to trigger a visual warning whenever the probe tip was within 30.48 cm (1 ft) of the test sphere (see Fig. 18) as measured by PROTOCOL.

Four iterations of the latency test were performed and the three constituent latencies making up overall latency were recorded as shown in Table 5. TCP/IP socket connections were used to transmit pose data from the iGPS server to PROTOCOL's input stage. Due to the high frequency of streaming data, it was critical to ensure that time

Table 5 Latency test results for PROTOCOL

Iteration #	Incoming external latency (ms)	Internal latency (ms)
1	15.4	13
2	8.0	20
3	3.3	18
4	8.1	10

The outgoing external latencies were negligible and are not shown

durations of incoming, internal, and outgoing segments were computed accurately by recording time stamps along with the pose coordinate for every packet of data. Thus, an individual pose coordinate's progress from the iGPS server through the entire visualization system could be tracked without error. Incoming external latency was measured as the time offset between a pose data point being streamed by the iGPS server and being received by the visualization system. Internal latency was measured as the time between a data point being received and its proximity analysis being completed. Outgoing external latency was measured as the difference in time stamps between analysis being completed (and warning being detected) and the visualization system acknowledging the warning condition by altering screen color from white to red.

In addition to the above, the iGPS system has a latency associated with itself due to the frequency at which it updates readings. The iGPS system has an update rate of 40 Hz which implies that a data point transmitted by it has a latency of 25 ms associated with it. The incoming external latency values presented in Table 5 do not include the 25 ms latency. During the experiments, the probe was being moved at an average speed of 0.54 m/s. As the iGPS system updates the location at 40 Hz, the probe tip's reported position can be up to 13.5 mm behind its actual position in the real world. For a given sensor and its corresponding dynamically tracked entity, the offset would increase with greater speed of motion and/or a reduction in the position update frequency. To account for this offset between reported and actual positions, the future work of the authors includes a stochastic approach to predict warnings in advance as introduced in the Conclusions and Future Work section.

The Nagle algorithm which coalesces smaller data packets into fewer larger packets is understood as being one of the causes for transmission delay in the case of incoming external latency [38]. Investigation of the data stream from the server application showed that a similar coalescing of data points was occurring with multiple data points having the same timestamp. As a result of multiple data points being sent in bursts, the time offset for some of the incoming external latencies was observed to be very high and those samples were ignored for the purpose of mean and standard deviation calculations. In the current implementation of the visualization system, the incoming sensor stream updates are associated with the update traversals of the overall visualization system.

As a result, the sensor stream data have some additional latency due to residing in the client socket's buffer prior to being processed. A modification of the data transmission

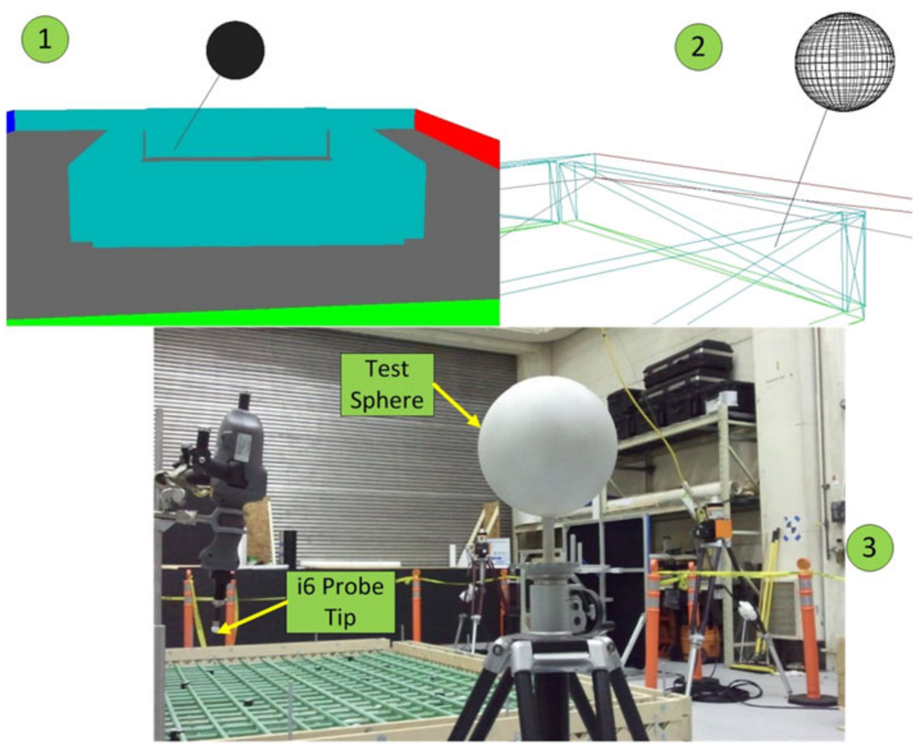

Fig. 19 *1* Real-time 3D visualization of experiment; *2* polygon surface models used for graphical analysis; *3* experiment setup with test sphere and i6 probe tip

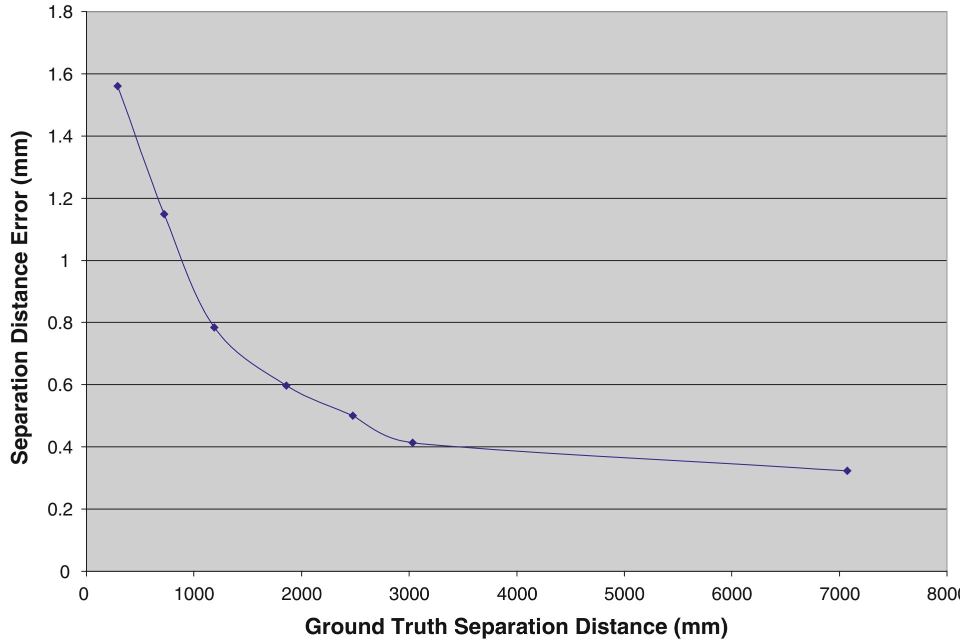

Fig. 20 Separation distance error as a function of separation distance

design to reduce latency for server and client ends is part of the authors' ongoing research. Internal latency times showed a relatively consistent value for analyzing the graphical database consisting of the test sphere and the probe tip as seen in Fig. 19(1, 2). The outgoing external latency was negligible since the 3D visualization itself was used to provide visual warnings and is thus not represented in the Table 5 results.

7.3 Discussion of the experimental results

The distance test results show that the mean separation distance error is 0.761 mm with a standard deviation of 0.446 mm. Figure 20 shows a plot of the separation distance error as a function of ground truth separation distance. The trend of increasing error as the separation distance decreases is potentially due to the position of the iProbe during the tests. In the case where the test sphere and iProbe were in very close proximity, the receivers on the iProbe were partially blocked by the test sphere resulting in small variances of the reported position. As the reported position of the iProbe tip's center is used to place its corresponding 3D model, variances of the reported positions led to unintended translations in the 3D model.

The latency test showed that the mean incoming external latency was 8.7 ms with a standard deviation of 4.99 ms. Mean internal latency was 15.25 ms with a standard deviation of 4.57 ms. The observed values of incoming external and internal latencies yielded frame rates in the range of 15–24 frames per second (fps) or Hz as observed in the OpenSceneGraph-based visualization. The frame rate did not yield any noticeable lag in operation and also provided fps values which lie in the minimum acceptable range of 15–30 Hz for real-time rendering giving smooth continuous motion [4, 35, 60].

8 Conclusions and future work

In this paper, the authors investigated the types of spatial conflicts that occur between entities on construction job sites and identified limitations in current methods in dealing with scenarios that involve concealed or buried infrastructure. A computing framework based on position–orientation sensor input and 3D virtual models that provides distance, collision, and tolerance queries is presented. The computation framework is designed to support a real-time 3D visualization scheme that provides equipment operators with information and job-site views that are not possible through conventional on-site or on-board monitoring cameras.

Results of proximity measurement performance and latency tests carried out at the NIST's ISAT were described. Evaluation of the suitability of the proximity monitoring framework is based upon its primary intended use, i.e., currently enforced regulations regarding minimum permissible distance between an excavator end effector and buried utilities. Best practice and safe excavation guidelines for excavator operators in many states require that no mechanized equipment be used within 0.6096–0.4572 m (2–1.5 ft) of utility extremities [1, 39, 40, 59]. The separation distance error achieved in the experiments is between 0.12 and 0.16 % of the minimum permitted distance, and is thus found to have acceptable accuracy for its intended application.

The latency values obtained yielded frame rates in the range of 15–24 fps. As the frame rates obtained did not introduce any noticeable lag and were found to be within the commonly accepted ranges for real-time rendering [4, 35, 60], it can be concluded that the proximity monitoring framework successfully achieves acceptable latency values. When combined with an analytical approach such as the one described in Liang and Lu [31], the presented framework can be used to achieve efficient real-time 3D visualization of articulated construction equipment.

The future goals in this research direction are to introduce a stochastic model in the warning mechanism. The current implementation is a reactive approach that warns users of impending collisions when entities cross safety thresholds in the real world. However, the impact of data transmission times from sensors to the on-board computation center can result in unfavorable latency. This can be averted through a predictive approach in which an object's current motion trajectory and speed are used to project when a safety threshold will be breached and warnings can be given further in advance without being affected by high transmission times to the same extent.

Acknowledgments The presented research was funded by the US National Science Foundation (NSF) via Grants CMMI-927475 and CMMI-1160937. The writers gratefully acknowledge NSF's support. Any opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the NSF, NIST, or the University of Michigan.

References

1. Alagasco (2012) Know what's below—always call 811 before you dig. http://blog.alagasco.com/know-whats-below-always-call-811-before-you-dig/. Accessed 06 Sep 2012
2. Arditi D, Ayrancioglu M, Shu J (2005) Worker safety issues in nighttime highway construction. Eng Constr Archit Manag 12(5):487–501
3. Bajaj C, Wyvill B, Bloomenthal J, Guo B, Hart J, Wyvill G (1996) Implicit surfaces for geometric modeling and computer graphics. SIGGRAPH Lecture 1996. http://www.cs.princeton.edu/courses/archive/fall02/cs526/papers/course11sig96.pdf. Accessed 17 June 2012
4. Bowman D (2008) Real-time 3D graphics for VEs. http://courses.cs.vt.edu/~cs5754/lectures/rtgraphics.pdf. Accessed 26 Jul 2012
5. Bryson S (1993) Effects of lag and frame rate on various tracking tasks. In: Proceedings of SPIE stereoscopic displays and applications, vol 1915
6. Bureau of Labor Statistics (2012) Census of fatal occupational injuries. http://www.bls.gov/iif/oshcfoi1.htm#2009. Accessed 17 Jun 2012
7. Chen JYC, Haas EC, Barnes MJ (2007) Human performance issues and user interface design for teleoperated robots. IEEE Trans Syst Man Cybern Part C Appl Rev 37:1231–1245
8. Cheng T, Teizer J (2011) Crane operator visibility of ground operations. In: Proceedings of the 28th ISARC, 2011, Seoul, Korea, pp 699–705
9. Cheng T, Venugopal M, Teizer J, Vela PA (2011) Performance evaluation of ultra wideband technology for construction resource location tracking in harsh environments. Autom Constr 20(8):1173–1184
10. Chi S, Caldas C, Gong J (2008) A crash avoidance framework for heavy equipment control systems using 3D imaging sensors. ITCON 13:118–133
11. Cho Y, Haas C, Liapi K, Sreenivasan SV (2002) A framework for rapid local area modeling for construction automation. Autom Constr 11(6):629–641
12. Dana PH (2000) Unit 017—global positioning system overview. http://www.ncgia.ucsb.edu/giscc/units/u017/u017.html. Accessed 17 Jun 2012
13. Dasgupta S, Banerjee A (2006) An augmented-reality-based real-time panoramic vision system for autonomous navigation. IEEE Trans Syst Man Cybern Part A 36(1):154–161
14. Durikovic R (2012) Lecture 13: representation of solids. http://www.sccg.sk/~durikovic/classes/MRT/3DModeling3%20Representation.pdf. Accessed 17 Jun 2012
15. Federal Highway Administration (FHWA) (2012) Worker safety. http://www.ops.fhwa.dot.gov/wz/workersafety/index.htm. Accessed 17 Jun 2012
16. Foley JD, van Dam A, Feiner SK, Hughes J (1992) Computer graphics: principles and practice, 2nd edn. Addison Wesley, Reading, pp 533–562
17. Geokov (2012) UTM—Universal Transverse Mercator Projection. http://geokov.com/Education/utm.aspx. Accessed 17 Jun 2012
18. Gottschalk S, Lin MC, Manocha D (1996) Obbtree: a hierarchical structure for rapid interference detection. In: Proceedings of SIGGRAPH'96, 1996
19. Hearn D, Baker MP (1994) Computer graphics, 2nd edn. Prentice Hall, Englewood Cliffs, pp 355–362
20. Hine III BP, Stoker C, Sims M, Rasmussen D, Hontalas P, Fong TW, Steele J, Barch D, Andersen D, Miles E, Nygren E (1994) The application of telepresence and virtual reality to subsea exploration. In: The 2nd workshop on mobile robots for subsea environments, Proceedings of ROV'94, May 1994
21. Hinze JW, Teizer J (2011) Visibility-related fatalities related to construction equipment. Saf Sci 49(2011):709–718
22. Huang X, Bernold L (1997) CAD integrated excavation and pipe laying. J Constr Eng Manag 123(3):318–323
23. Huber D, Herman H, Kelly A, Rander P, Warner R (2009) Real-time photorealistic visualization of 3D environments for enhanced teleoperation of vehicles. In: Proceedings of the 2nd international conference on 3D digital imaging and modeling, Kyoto, Japan
24. Hwang S (2011) Ultra-wideband technology experiments for real-time prevention of tower crane collisions. Autom Constr 22:545–553
25. Kamat VR, Martinez JC (2005) Dynamic 3D visualization of articulated construction equipment. J Comput Civil Eng 19(4):356–368
26. Kamat VR, Martinez JC (2007) Interactive collision detection in three-dimensional visualizations of simulated construction operations. Eng Comput 23(2):79–91
27. Kim CW, Haas CT, Liapi KA, Caldas CH (2006) Human-assisted obstacle avoidance system using 3D workspace modeling for construction equipment operation. J Constr Eng Manag 20(3):177–186
28. Kini AP, King K, Kang S (2009) Sensor calibration and real-time tracking of a backhoe loader using the iGPS system. Qual Dig Mag. http://www.qualitydigest.com/inside/cmsc-article/sensor-calibration-and-real-time-tracking-backhoe-loader-using-igps-system.html. Accessed 17 June 2012
29. Kwon S, Bosche F, Kim C, Haas CT, Liapi K (2004) Fitting range data to primitives for rapid local 3D modeling using sparse range point clouds. J Autom Constr 13(1):67–81
30. Larsen E, Gottschalk S, Lin MC, Manocha D (1999) Fast proximity queries with sphere swept volumes. Technical report TR99-

018, Department of Computer Science, UNC Chapel Hill. http://gamma.cs.unc.edu/SSV/ssv.pdf. Accessed 22 June 2011
31. Liang X, Lu M (2012) Real-time 3D positioning and visualization of articulated construction equipment: case of backhoe excavators. In: Proceedings of the international workshop on computing in civil engineering. ASCE, Clearwater Beach
32. Lin MC, Gottschalk S (1998) Collision detection between geometric models: a survey. ftp://cs.unc.edu/pub/users/lin/cms98.pdf. Accessed 19 Nov 2011
33. Lodha S, Charaniya AP, Faaland NM, Ramalingam S (2002) Visualization of spatiotemporal GPS uncertainty within a GIS environment. In: Proceedings of SPIE conference on radar sense technology and data visualization, pp 216–227, April 2002
34. MacCollum DV (1995) Construction safety planning. Wiley, New York, pp 53–54
35. Malhotra P (2002) Issues involved in real-time rendering of virtual environments. Master's thesis, Faculty of Virginia Polytechnic, 2002
36. Marks E, Teizer J (2012) Safety first: real-time proactive equipment operator and ground worker warning and alert system in steel manufacturing. Iron Steel Technol AIST 9(10):56–69
37. McHenry K, Bajcsy P (2008) An overview of 3D data content, file formats and viewers. Technical Report ISDA08-02, 2008
38. Nagle J (1984) Congestion control in IP/TCP internetworks, RFC-896. Internet Eng Task Force. http://tools.ietf.org/pdf/rfc896.pdf. Accessed 26 Jul 2012
39. National Fuel (2012) National fuel reminds customers to call before you dig. http://www.natfuel.com/news/04-25-12%20Call%20Before%20You%20Dig%20-%20WEB4252012-123657.pdf. Accessed 06 Oct 2012
40. New Jersey One Call (2011) A guide to safe excavation practices in New Jersey. http://www.pseg.com/business/local_government/safety/pdf/booklet.pdf. Accessed 06 Sep 2012
41. Nikon Metrology Inc. (2012) iGPS. http://www.nikonmetrology.com/en_US/Products/Large-Volume-Applications/iGPS/iGPS/(brochure). Accessed 17 Jun 2012
42. PQP Gamma (1999) PQP—a proximity query package. http://gamma.cs.unc.edu/SSV/. Accessed 22 Jun 2011
43. Occupational Safety and Health Administration (2010). http://www.osha.gov/pls/oshaweb/owadisp.show_document?p_table=INTERPRETATIONS&p_id=27308. Accessed 17 Jun 2012
44. Oloufa A, Ikeda M, Hiroshi O (2002) GPS-based wireless collision detection of construction equipment. In: Proceedings of the 19th international symposium on automation and robotics in construction, ISARC. National Institute of Standards and Technology, Gaithersburg, pp 461–466
45. PHMSA (2012) Significant pipeline incidents through 2010 by cause. http://primis.phmsa.dot.gov/comm/reports/safety/SigPSIDet_2001_2010_US.html?nocache=9140#all. Accessed 11 Jan 2012
46. Preston R, Doane J, Kiwio D (2005) Using real world data to create OPNET Models DRAFT. The MITRE Corporation. http://www.mitre.org/work/tech_papers/tech_papers_05/05_0850/05_0850.pdf. Accessed 17 Jun 2012
47. Ruff TM, Hession-Kunz D (2001) Application of radio-frequency identification systems to collision avoidance in metal/nonmetal mines. IEEE Trans on Ind Appl 37(1):112–116
48. Schiffbauer WH, Mowrey GL (2001) An environmentally robust proximity warning system for hazardous areas. In: Proceedings of the ISA emerging technologies conference, Triangle Park, NC, vol 2091, p 10
49. Shen C, O'Brien JF, Shewchuck JR (2004) Interpolating and approximating implicit surfaces from polygon soup. ACM Trans Graph 23(3):896–904
50. Son H, Kim Ch, Kim H, Choi K-N, Jee J-M (2008) Real-time object recognition and modeling for heavy-equipment operation. In: Proceedings of the 25th international symposium on automation and robotics in construction, Vilnius, 2008, pp 232–237
51. Son H, Kim C, Choi K (2010) Rapid 3D object detection and modeling using range data from 3D range imaging camera for heavy equipment operation. Autom Constr 19(7):898–906
52. Talmaki S (2012) Real-time visualization for prevention of excavation related utility strikes. Ph.D. Dissertation, Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI
53. Talmaki S, Kamat V (2012) Real-time hybrid virtuality for prevention of excavation related utility strikes. J Comput Civil Eng. doi:10.1061/(ASCE)CP.1943-5487.0000269
54. Teizer J, Kim C, Haas CT, Liapi KA, Caldas CH (2005) A framework for real-time 3D modeling of infrastructure. Transp Res Rec J Transp Res Board 1913:177–186
55. Teizer J, Caldas CH, Haas CT (2007) Real-time three-dimensional occupancy grid modeling for the detection and tracking of construction resources. ASCE J Constr Eng Manag 133(11):880–888
56. Teizer J, Allread BS, Mantripragada U (2010) Automating the blind spot measurement of construction equipment. Autom Constr 19(4):491–501
57. Teizer J, Allread BS, Fullerton CE, Hinze J (2010) Autonomous pro-active real-time construction worker and equipment operator proximity safety alert system. Autom Constr 19(5):630–640
58. Teschner M, Heidelberger B, Manocha D, Govindaraju N, Zachmann G, Kimmerle S, Mezger J, Fuhrmann A (2005) Collision handling in dynamic simulation environments. In: Eurographics Tutorials, 2005, pp 79–185
59. Virginia State Corporation Commission (2005) Exposing underground utility lines. http://www.scc.virginia.gov/urs/mutility/docs/exp_bp.pdf. Accessed 06 Sep 2012
60. Wloka MM, Zeleznik RC (1996) Interactive real-time motion blur. Vis Comput 12(6):283–295
61. Zhang C, Hammad A, AlBahnassi H (2008) Collaborative multi-agent systems for construction equipment based on real-time field data capturing. In: Next generation construction IT: technology foresight, future studies, road mapping, and scenario planning, vol 14, pp 204–228