A CRASH AVOIDANCE FRAMEWORK FOR HEAVY EQUIPMENT CONTROL SYSTEMS USING 3D IMAGING SENSORS

SUBMITTED: May 2007
REVISED: September 2007
PUBLISHED: April 2008
EDITORS: B. Akinci and C. Anumba

Seokho Chi, Ph.D. Candidate
Department of Civil, Architectural, and Environmental Engineering, University of Texas at Austin, USA
email: shchi@mail.utexas.edu

Carlos Caldas, Assistant Professor
Department of Civil, Architectural, and Environmental Engineering, University of Texas at Austin, USA
email: caldas@mail.utexas.edu

Jie Gong, Ph.D. Candidate
Department of Civil, Architectural, and Environmental Engineering, University of Texas at Austin, USA
email: jie.gong@mail.utexas.edu

SUMMARY: *This paper presents a preliminary crash avoidance framework for heavy equipment control systems. Safe equipment operation is a major concern on construction sites since fatal on-site injuries are an industry-wide problem. The proposed framework has potential for effecting active safety for equipment operation. The framework contains algorithms for spatial modeling, object tracking, and path planning. Beyond generating spatial models in fractions of seconds, these algorithms can successfully track objects in an environment and produce a collision-free 3D motion trajectory for equipment.*

KEYWORDS: *crash avoidance, path planning, spatial modeling, object tracking.*

1. INTRODUCTION

Heavy equipment operation is fundamental to construction projects. Since equipment operation can directly affect project performance, much research has been conducted to develop efficient methods for equipment control. Computer-assisted systems now allow equipment operators to improve work performance, increase productivity, and reduce operating costs more than ever. However, these systems do not fully address construction safety concerns. Many on-site accidents and fatalities are the result of unsafe equipment operation; the large scale of the equipment and the changeable nature of construction sites combine to create unpredictable and dangerous situations (Kim et al, 2006). Among a total of 1,186 fatal work injuries that occurred in the U.S. in 2005, about 25% of the total injuries were equipment-related (Bureau of Labor Statistics, 2005). Moreover, in recent decades the number of employed skilled operators has decreased due to outdated training and early retirement (Business Roundtable, 1997). All of these factors have contributed to the increased demand for automated safety features, such as crash avoidance methods.

The primary objective of the research presented in this paper is to develop a preliminary framework for crash avoidance that in the future can be incorporated in heavy equipment control systems. The proposed framework can monitor job sites in real-time by utilizing three-dimensional image sensing technology. By doing that it can anticipate possible collisions between heavy equipment and on-site objects or people, as well as facilitate the development of safe operation plans that help the equipment operator avoid crashes.

The proposed crash avoidance framework consists of three major algorithms: a 3D spatial modeling algorithm, an object tracking algorithm, and a 3D path planning algorithm. The 3D spatial modeling algorithm can extract spatial information from the local environment in real-time and use it to construct a 3D model (Teizer, 2006). The object tracking algorithm is able to compare sequential built models in fractions of seconds. The 3D path planning algorithm can calculate shortest collision-free paths that mobile equipment can take and the safest motion sequences for manipulator equipment. Experiments in a laboratory testbed using a research-prototyped mobile robot and a computer simulation were used to verify the feasibility of the proposed framework.

The remainder of the paper is organized as follows. In Section 2, a review of the related literature on path planning, object tracking, and image matching is included. In Section 3, the proposed framework for crash avoidance is described with detailed explanations of contained algorithms. In Section 4, experimental results are shown. Finally, Section 5 concludes the paper, providing result analyses, discussing research contributions, and proposing future research directions.

2. BACKGROUND AND LITERATURE REVIEW

2.1 Path planning for automation on construction sites

Much path planning research has been conducted to develop an efficient automated process for heavy construction equipment operation. Previous path planning studies can be broadly divided into three categories: (1) motion planning for a manipulator of heavy equipment placed in a static position, (2) trajectory planning for mobile heavy equipment, and (3) fully-automated path planning for automated heavy equipment operation. This section will review the contributions and limitations of previous path planning studies.

First of all, motion planning for manipulator equipment has been explored by many research teams. Stentz et al. (1999) examined autonomous truck loading and developed software for excavator loading at fixed equipment positions. The software was capable of determining operation paths that extended from soil digging to dumping soil into a truck, and allowed the truck to detect and avoid surrounding obstacles. Besides addressing the loading operation, some researchers have delved into crane operation. Reddy and Varghese (2002) developed search algorithms to optimize lift paths from picking to placing for single crane operation. Ali et al. (2005) investigated the motion paths of cooperative cranes to address risks related to plan preparation and lift execution. These researches developed a technology that allowed operators to quickly produce reliable and short lift plans. In spite of their contributions, these studies can be extended. Such path planning is partly-automated; it plans equipment manipulators' motion paths only in their static positions. Therefore, if the equipment is itself moving, the motion path calculation for manipulator operation might become unfeasible. Also, these systems can not be applicable to general heavy equipment use because they were designed only for specific equipment types.

Path planning for mobile heavy equipment has also been investigated with the aim of planning a shortest and accident-free travel path from a starting position to a goal position (Miller et al, 1998). First of all, path planning for earthmoving activities was studied by some researchers. Tserng et al. (2000) developed an automated landfill system for multiple-truck and multiple-compactor operations. This system simulated the landfill processes in the pre-planning phase and was able to find safe motion patterns for autonomous landfill equipment in the construction phase. Kim et al. (2003) explored methods for determining an efficient travel path for earthmoving construction equipment on unpredictable and dynamic construction sites. In addition to earthmoving-related path planning, path planning studies in general have been performed for mobile heavy equipment operation. Lee and Adams (2004) developed a path planning model for the operation of multiple mobile construction equipment simultaneously working on construction sites, and Soltani and Fernando (2004) investigated path planning based on multiple-objective analyses (e.g.: cost, safety, and visibility) to generate shortest trajectory calculation. To help site planners predict uncertainty and subjectivity of equipment movement, they examined not only the paths of materials on a construction site but also on-site heavy equipment movement. Notwithstanding these accomplishments, there are improvement opportunities in existing path planning research for mobile heavy equipment operation. First, the previous research mostly makes use of pre-determined knowledge. Though it offers advantages for pre-planning, it is not suited for real-time applications. In addition, the majority of algorithms used in these earlier studies have difficulty reflecting complicated spatial site conditions since they are mostly 2D-based.

Yet fully-automated path planning can overcome many of the challenges faced by the partly-automated approaches discussed above. Saeedi et al. (2005) developed a vision-based excavator control system that successfully controlled excavator movement and its slippage. It was capable of three-dimensionally planning an equipment manipulator's motion route in the work space. Similarly, Sarata et al. (2006) developed an autonomous loading operation system for a mobile wheel loader. Their system was able to perform environmental modeling, plan loader travel trajectory, and control loader scooping motion. Fully-automated processes provide can provide these benefits for equipment operation, but at the same time uncertainties remain for their applications on construction sites; the harsh and unpredictable nature of construction sites as well as the high system cost of these systems makes it difficult to guarantee their performance (Kim et al, 2006).

2.2 Object tracking

Analyzing spatial information is crucial to path planning since possible paths vary depending on spatial information such as geometry, position, and motion characteristics of objects. In spatial information analyses, the ability to track objects is paramount. The general object tracking process involves first determining an object's motion information such as velocity vector or acceleration vector. Then the process determines whether the object is static or moving. Thereafter, it tracks moving objects, and predicts their future motion routes using data taken from their dynamic context. For these reasons, object tracking is a fundamental step for an effectual path planning process.

Several researchers have studied asset detection and monitoring for construction safety based on tracking technologies such as Radio-frequency Identification (RFID), Global Positioning System (GPS), and Ultra-Wideband (UWB). Schiffbauer and Mowrey (2001) used RFID technology to improve work zone safety by warning workers when they were close to heavy equipment. Oloufa et al. (2002) developed and implemented a collision detection system based on GPS technology for tracking heavy construction equipment and transmitting location information to a central server. Abderrahim et al. (2005) tracked the positions of workers and machines using miniature positioning and communication instruments. Riaz et al. (2006) developed a health and safety management system using a combination of GPS, RFID, and wireless networks. Their system tracked operators, workers, and plants for the purpose of reducing vehicle/pedestrian collisions. Teizer and Castro-Lacouture (2007) emphasized on advantages of UWB for rapid communication and precision localization on job sites. However, these studies also revealed some shortcomings of tracking technologies: (1) tagging and tracking individuals can create personal privacy issues, (2) RFID- and UWB-based tracking require a large number of tags and antennas to monitor and protect all resources on the site, (3) RFID- and UWB-based technologies track the tags attached to construction components and equipment; for large objects attaching just one tag to it is usually not enough to track and protect all of their boundaries and (4) there are several technical limitations that need to be addressed when implemented these technologies on realistic construction environments. Examples of these technical limitations include the following: radio signals can be affected by interference from multiple-path effects and other radio signal sources (Ruff, 2000), GPS technology still has capability gaps in its measuring accuracy (Riaz et al, 2006), and UWB network is sensitive to errors caused by reflection from metal obstacles or produced by a digital compass when interrupted with surrounded magnetic objects (Williams et al, 2007). Thus, it is important to explore alternative solutions, including tag-less approaches for object tracking.

Much object tracking research has been conducted in the field of computer vision. Past object tracking studies can be generally divided into two categories: feature-based object tracking and model-based object tracking (Zhang and Faugeras, 1992, Jung and Wohn, 1998, Marchand et al, 2001). The feature-based object tracking approach generates estimates of feature positions in sequential image frames so that the real motion of objects can be analyzed (Zhang and Faugeras, 1992). This approach first extracts features of interest, such as points, lines, curves, surfaces, or volumes, from sequential image frames, and then finds correspondences between them. Because this simple motion tracking capability is based on the assumption that each extracted feature in a frame is independent from any others, this approach does not require high-level knowledge of an object's geometry (Jung and Wohn, 1998). In spite of this strength, the feature-based tracking approach also has some limitations. It has difficulty generating a reliable feature from an image when the motion of an object is complicated; such complicated motion can occur when an object is moving unpredictably and quickly within an obstacle-cluttered construction job site, or when some parts of the image are obscured or distorted (Jung and Wohn, 1998). In these cases, finding resemblance between two extracted features seems to be meaningless. Moreover, it is difficult to determine a scale factor with the feature-based approach (Zhang and Faugeras, 1992). Because this approach does not employ considerable *a priori* feature information, consistent feature comparison is impossible when it is used. The problems with this tracking approach has contributed to the increasing demand for an *a priori* model database, that would able to prevent false tracking due to noise disturbance and eventually improve tracking efficiency.

Unlike the feature-based approach, the model-based object tracking approach uses *a priori* knowledge of geometric object models to recognize an object's shape and appearance (Polat et al, 2003). Although the feature-based tracking approach directly compares two features obtained from sequential image frames, the model-based approach can compare an extracted feature in each frame to pre-determined models. Because of its potential for improving object tracking, the model-based object tracking has generated considerable research interest. Rehg and Kanade (1995) tracked articulated hand motion by exploiting a kinematic model of finger shapes, and Dellaert et al. (1998) looked into a Kalman-filter-based tracking approach by using texture mapping as the measurement model. Similarly, Nickels and Hutchinson (2001) developed a method for tracking complex and articulated objects based on observation of a monocular grayscale image of the scene. This method built a data set with an appearance model and a kinematic structure of the objects. In addition, Marchand et al. (2001) exploited a CAD model by transforming the projection of the CAD model of the object into the spatial intensity gradients.

Three-dimensional model-based object tracking has been also examined by some researchers since a 3D approach is capable of improving the accuracy and robustness of feature correspondences as well as more pertinently representing real objects than a 2D approach can. Koller et al. (1993) tracked different types of cars based on a remarkably effective parameterized 3D car model. Using recorded road traffic scenes, they were able to automatically track moving vehicles. Also, Gavrila and Davis (1996) successfully tracked unconstrained human movement tracking dancers' motion by means of a large database of human actions.

The 3D model-based tracking approach first uses pre-determined model templates that are made up of a 3D rendered model with pose information of the rendered model (Polat et al, 2003). From the model template data, this approach estimates the position and pose parameters of the feature extracted from the image frame. Then it matches the image feature to given models to determine motion. Thus, extracting the most pertinent features from an image and accurately matching those features to a model are significant concerns for achieving effective 3D model-based tracking. These concerns can be addressed by employing an efficient image matching method that will be introduced in the next section.

2.3 Image matching

Image matching algorithms are designed to find correspondences between templates and given portions of images (Zhijia et al, 2003). A considerable amount of research has been conducted on image matching and has generally been divided into two categories: area-based image matching and feature-based image matching (Cochran and Medioni, 1992, Dowman, 1998, Wei et al, 1998, Zhijia et al, 2003). The area-based image matching approach considers intensity or texture of the image pixels (Dowman, 1998). Beginning with the assumption that the disparities of pixel intensity in each frame are constant, this matching approach finds the corresponding pixel in the other image (Wei et al, 1998). Rziza et al. (2000) examined segmentation of stereoscopic images to match images based on a sense disparity map, and Chambon and Crouzil (2004) investigated correlation-based matching algorithms that behave efficiently when objects are occluded.

The above studies were able to efficiently generate direct dense disparity maps (Cochran and Medioni, 1992), yet the area-based matching algorithm was found to have some shortcomings. First, it had difficulty selecting an appropriate area size that not only could include texture variation for matching but also could avoid the effects of projective distortion (Wei et al, 1998). This area scale issue has prompted more focused image matching research. Kanade and Okutomi (1994) were able to determine area size by minimizing the uncertainties of disparity estimates, and Koo and Jeong (2001) developed a method for adapting the area size to local intensity variation. Nevertheless, they could not avoid the estimation errors that resulted from the assumption of constant disparity (Wei et al, 1998). Also, noise became prevalent when there was a lack of texture. These drawbacks suggest that abstract features are more conducive to precise matching than the texture regions since features generate less noise and can be straightforwardly matched when the texture regions are less textured (Cochran and Medioni, 1992).

The feature-based image matching approach uses symbolic descriptions of images in order to attain higher accuracy correspondences (Dowman, 1998). This approach extracts features such as points, line segments, or a combination of regions, lines, and edges. These acquired features are then used as matching primitives. A significant amount of research on this matching approach has been performed. Marapane and Trivedi (1994) examined steady overall matching system integrating regions, lines, and edges. Zhou and Shi (2002) developed a new feature point matching algorithm that was able to effectively eliminate outliers and acquire a high ratio of correct matches. Schwarz and Lobo (2005) investigated segment-based matching for hand pose estimation.

To acquire 3D spatial information, the proposed framework utilizes a high-frame-rate range sensor that provides a matrix that stores a number of distance data points. One of the most pertinent feature-based matching approaches is feature point matching; this approach extracts represented points from an image and compares theses meaningful points with points of model shapes (Zhou and Shi, 2002). Thus, the feature point matching seems most applicable to the framework using such sensor. However, extracting points from the image with this approach can be time consuming since the high-frame-rate sensor derives a dense point cloud from the overall volume of a scanned image. Therefore, the research team sought a reasonable method for using the entire set of obtained dense point data at fast computational speeds.

2.4 Background and literature review summary

The background and literature reviews revealed the gaps in earlier research on topics related to active safety control systems for construction equipment. The approaches currently available for manipulator equipment operation are not generally applicable to heavy equipment since most systems are designed only for specific equipment types. For instance, truck loading path planning for excavator operation can only be used for loading operations. Conversely, mobile equipment trajectory planning cannot be adapted to manipulator equipment operation. Most methods for mobile equipment trajectory planning are generally suitable for pre-planning process, not real-time applications; these systems have a limited capacity for resolving safety issues on job sites. Moreover, these 2D-based systems are unable to reflect complicated spatial site conditions such as the 3D open space under a crane or under a bridge.

Fully-automated operating systems are not economically feasible and are not easily applied to existing heavy equipment. Therefore, there is a need to develop generic crash avoidance algorithms that can easily be applied to existing heavy equipment. These algorithms should consider both mobile equipment trajectory planning and equipment manipulator motion planning, and they should also support real-time 3D-based assistance for equipment operators.

3. FRAMEWORK FOR CRASH AVOIDANCE

Fig. 1 shows an overview of the proposed framework for crash avoidance. The crash avoidance framework consists of four major processing phases: (1) data acquisition, (2) 3D spatial modeling, (3) object tracking, and (4) 3D path planning. To facilitate the data acquisition process, the proposed framework exploits a high-frame-rate range sensor and acquires spatial information of a local environment from such a sensor’s field of view. An occupancy grid based modeling algorithm (Teizer, 2006) then transforms obtained spatial data into a 3D grid map. A 3D local model is built based on the data from occupied cells, and at the same time a grid-based clustering algorithm (Tan et al, 2005) represents objects by grouping the occupied cells. Such object clusters can be tracked by employing the Hausdorff image matching algorithm (Olson, 1998). This algorithm compares positions of object clusters in sequential image frames and tracks objects in the 3D environmental model. Finally, a 3D path planning algorithm can generate a safe motion path not only for manipulator equipment operation but also for mobile heavy equipment operation. More details on algorithms will be covered in the next sections.

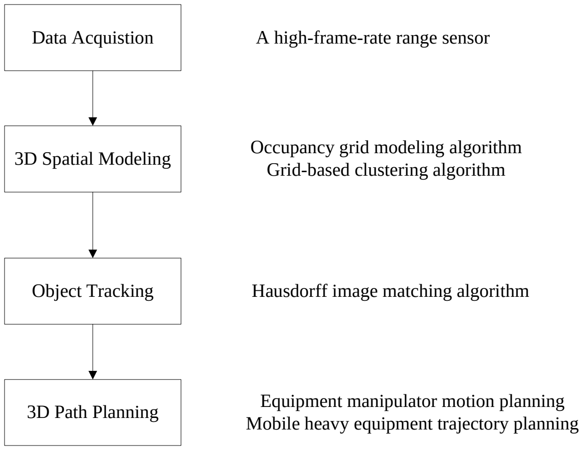


FIG. 1: Framework for crash avoidance.

3.1 Real-time 3D spatial modeling

The real-time spatial modeling algorithm models the 3D local environment by analyzing acquired spatial data. The proposed research used a method already developed by the research team (Teizer, 2006). An occupancy grid modeling algorithm (Moravec and Elfes, 1985) divides a local environment into a 3D grid space. The grid space is made up of a number of occupancy cells and each occupancy cell holds a probability value that the cell is occupied. Acquired range data can be plotted into the occupancy cells. When a range value is plotted into a cell, a cell value increases. After all range data are updated into the cells, cell occupancy can be determined. To make this determination, a pre-determined probabilistic threshold value is used. If the value of an occupancy cell is larger than the threshold value, the algorithm considers the cell to be occupied and gives it a value of 1. Conversely, if the cell value is smaller, the cell is considered an empty cell and gives it a value 0. The algorithm retains occupied cell information for further processes.

There are two important considerations in the implementation of the algorithm; one is how to initialize a grid space, and the other is how to plot range data. For the grid space initialization, a proper occupancy cell size must first be set. If a cell size is small, the algorithm can model the local environment in a higher level of detail. However, more computation time is required for processing a number of occupied cell data for small cells. On the other hand, if a cell size is large, the algorithm can be processed quickly, but the results are less accurate. Therefore, it is crucial to define the best cell size for each specific situation with the aim of effectively representing an environment in real-time.

Next, when range data are plotted, the data can be plotted based on pixel resolution. A high-frame-rate range sensor typically provides a matrix that stores the distance information of the local environment. Each pixel of the matrix can only hold one range value and each pixel has a different resolution depending on the distance from the sensor to the object; a pixel size is large at a long distance, but small at a short distance. That is to say, the meaning of one range value in a large pixel is different from one in a small pixel. Therefore, range data need to be plotted into occupancy cells depending on the distance. When range values are plotted, more than one point (e.g. 2 points or 3 points) is plotted for a range value at a long distance, whereas less than one point (e.g. 0.5 point) is plotted for one at a short distance. After all range data are updated into occupancy cells with the same method, the occupancy of the cells is determined by the number of plotted points in each one.

Occupied cells can be grouped by a grid-based clustering algorithm (Tan et al, 2005). This clustering algorithm first generates clusters from groups of neighboring occupied cells and then conducts noise removal while eliminating low density clusters. The remaining clusters represent objects in the local environment, and clustering results become essential for the object tracking process.

3.2 Object tracking

An object tracking algorithm tracks object motion based on comparisons of position in sequential image frames. One of the biggest challenges to the object tracking process is data association; it is difficult to associate one object in the first frame with the same object in the second frame. The probabilistic Hausdorff image matching algorithm (Olson, 1998) can resolve this data association issue. This algorithm is one among the many feature point matching approaches and is based on the Hausdorff distance, a calculation that measures how closely each point of an image set lies near some point of a model set (Polat et al, 2003). Since the occupancy grid modeling algorithm extracts only occupied points out of a number of dense point clouds, the final number of points is relatively smaller than the original number of points. Thus, the Hausdorff matching algorithm can simply and quickly use overall extracted points without any data transformation. It reliably identifies objects by determining consistent partial matches between the extracted images and the models (Polat et al., 2003). The likelihood function for matching criterion *L(t)* is (Olson, 1998):

$$\log L(t) = \frac{\sum_{i=1}^{m} \log p(D_i;t)}{m} \qquad \textit{(Eq. 1)}$$

where, M = {μ_1, μ_2,…, μ_m} represents a set of model features,

I = {v_1, v_2,…, v_m} represents a set of image features,

$t \in T$ is a random variable describing the position of the objects, and

D is the distance from each model pixel to the closest image pixel.

Update log-likelihood by using the following equation:

$$\log p(D_i;t) = \begin{matrix} k_1 + k_2, & if & D_i \le \delta \\ k_1 & & otherwise \end{matrix} \qquad \textit{(Eq. 2)}$$

where, δ is a given distance error threshold. The precise k_1 and k_2 values are not important as long as $k_2 > 0$. The degree of resemblance between two images assigns constant probability log $p(D_i;t)$. Using the principle of maximum likelihood estimation—the larger the likelihood, the more closely the image matches the model—the most similar image can be found from the model set.

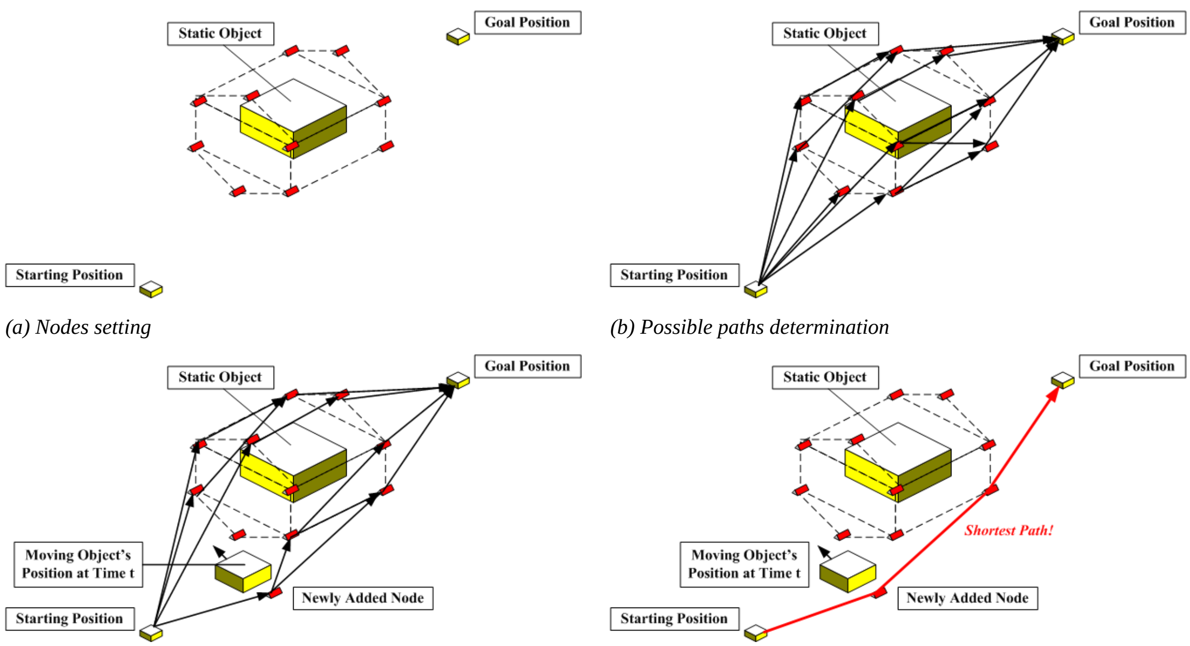
When this image matching algorithm is used, objects in sequential image frames can be tracked. In captured sequential frames, first a model set is built based on objects in the first frame. Objects in the second frame are then compared with the objects in the model set. By determining the extent of resemblance between objects in both frames, the algorithm is able to associate objects from one frame to the next (Huttenlocher et al, 1993). Thereafter, the algorithm determines objects' movement information based on position differences between image frames. If no position differences are detected between objects in sequential frames, they are classified as static objects. However, if there are any differences, they are tracked as moving objects.

3.3 3D path planning

Based on the motion characteristics of tracked objects—such as a static object's position and volume, and a moving object's initial position, volume, moving direction, and moving velocity—the proposed preliminary path planning algorithm calculates an optimized operation path for heavy equipment. For mobile heavy equipment operation, it generates a path from a starting position to an ending position. Similarly, for manipulator equipment operation, it determines a path from a picking position to a placing position. The research team investigated the algorithm development in previously conducted path planning research (Miller et al, 1998, Soltani et al, 2002, Wan et al, 2005). There are three major steps for path planning: (1) static object consideration to determine possible motion paths based only on the static environment, (2) moving object consideration to modify the paths developed in the first step in order to prevent collisions with moving objects, and (3) path optimization to calculate a shortest path among possible motion trajectories.

The path planning algorithm starts by taking static objects into account. The algorithm first sets task and interim nodes in the 3D local environment. The task nodes contain a beginning node at the starting or picking position and a stopping node at the ending or placing position. The interim nodes are set three-dimensionally around static objects. For the interim node setting, a safety margin is provided between a surface of a static object and a node in order to decrease the likelihood of a collision. This pre-determined safety margin is based on the equipment's size and motion volume. Fig. 2(a) shows the nodes set around a static object.

After the node setting process, possible discrete paths are determined. A discrete path is a straight path between any of the set nodes. The simplest discrete path would be a direct path from the beginning node to the stopping node. Since the discrete path can be blocked by static objects, ascertaining each path's availability is an essential step. To make this determination, the algorithm three-dimensionally compares the shortest distance between the path and the static object to the safety margin. If the distance is larger than the safety margin, the path is viable. However, if the distance is smaller, taking the path may result in a collision. As illustrated in the Fig. 2(a), the direct path from the starting position to the goal position is blocked by the static object. That is to say, the distance between the path and the object is zero. Therefore, because this motion sequence can create a collision state, the algorithm cannot generate such a path. Fig. 2(b) shows a set of possible discrete paths generated after such consideration.



(a) Nodes setting

(b) Possible paths determination

(c) Paths after moving object consideration

(d) Shortest path calculation

FIG. 2: 3D path planning

Next, the path planning algorithm considers moving objects in the local environment. Since a moving object can interrupt discrete paths already-determined by the previous step, the paths need to be revised to prevent possible collisions between machinery taking the path and the moving object. The algorithm compares a position on a path (a position of the mobile equipment or the manipulator equipment) at time *t* with a position of the moving object at time *t*. If the distance between the two positions at time *t* is smaller than the safety margin, a collision might occur. In this case, a new node is added to prevent moving object's entry into the path. Moreover, possible discrete paths should be re-calculated based on the revised node information. The algorithm repeats all of the above steps until all paths become accident-free paths. Fig. 2(c) shows modified final paths after the algorithm has considered moving objects.

Lastly, the path planning algorithm generates the shortest path. Any combination of all discrete paths can be a possible motion route. The algorithm considers all path combinations from the beginning node to the stopping node and calculates total travel distance of each motion route. After making these considerations, the algorithm can determine the shortest collision-free path with the lowest travel cost. Fig. 2(d) shows the calculated shortest path.

4. EXPERIMENTAL RESULTS

To assess the feasibility of the proposed crash avoidance framework, experiments and computer simulation were conducted by the research team. The team formulated three basic hypotheses before conducting the experiments: (1) the framework can be processed in real-time, (2) the framework can detect objects from the local environment and track their movement, and (3) the framework can predict possible collision state and provide a safe motion path for equipment. The preliminary experiments were executed in a laboratory setting using a research-prototyped mobile robot to represent mobile equipment operation, and the computer simulation was carried out to verify manipulator equipment operation. The experimental testbed was constructed in the Field Systems and Construction Automation Laboratory (FSCAL) at The University of Texas at Austin. Algorithm codes for 3D spatial modeling, object tracking and path planning were written with the C++ programming language. For the data acquisition process the team used a Swiss Ranger 2 (SR-2), a high-frame-rate range sensor developed by the Swiss Center for Electronics and Microtechnology (CSEM). The experiments for mobile equipment trajectory planning used a Pioneer 3-AT (P3-AT), a versatile all-terrain robotic platform developed by MobileRobots. The SR-2 was mounted on the P3-AT for implementation. In addition, Matlab software (version 7.0.0, R14) was used for visualizing the processed results, especially the results of the computer simulation for manipulator equipment motion planning.

4.1 Crash avoidance for mobile equipment operation

The primary purpose of the experiments was to examine the feasibility of the proposed crash avoidance framework by showing that the mobile robot successfully traveled without causing any collisions. Yet, in a real application, a control system based on the proposed framework would be incorporated into the heavy equipment to improve safety during its operation. The framework would be able to monitor potential hazards in the local environment in real-time.

4.1.1 Crash avoidance in a 2D environment

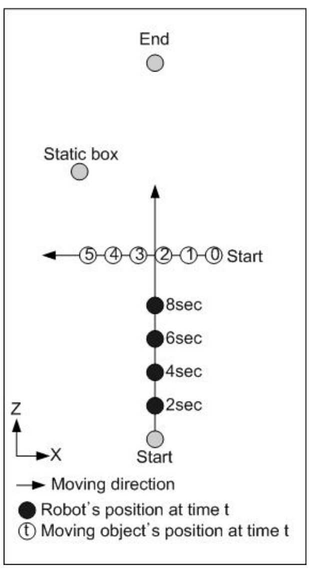
The initial experimental environment was comprised of (1) a box and (2) a concrete masonry unit (CMU) block mounted on a wire-controlled moving cart (Fig. 3). The box represented a static object and the CMU block represented a dynamic object. In order to examine how a local environment impacts path decision making, the experiments were conducted with the cart moving in diverse directions at different speeds. In addition, different initial speeds of the mobile robot were tested. The static box was placed at random locations. Among the experiments performed with different settings, two experiments will be described in this section. These provide a good illustration on how the motion characteristics of the moving objects were able to affect production of the final path.

In these two experiments, the CMU block was moved from the right to the left of the field. The starting position of the mobile robot was set at (0, 0, 0cm), and the ending position was set at (0, 0, 450cm). This (x, y, z) odometry represented a horizontal axis, a vertical axis, and a distance from the sensor to the object. The zero y value stood for the actual ground of the experimental field. The positive value signified a right direction on the horizontal axis and an upper direction on the vertical axis. Different movement conditions were set for the moving objects. The mobile robot's velocity was 20cm/sec in the first experiment and 40cm/sec in the second experiment. The forwarding speed of the moving object in the first experiment was faster than it was in the second experiment. The static box was placed behind the moving cart.

FIG. 3: Experimental setup.

In the first experiment, the SR-2 mounted on the mobile robot captured two images from the local environment. In the first image frame, the volume centroid value of the first cluster (the cluster of the CMU block) was set at (65, 50, 215cm) from the sensor position and the same value of the second cluster (the cluster of the box) was set at (-90, 40, 320cm). In the second frame, the volume centroid value of the first cluster was set at (55, 50, 215cm) and the same value of the second cluster was set at (-90, 40, 320cm). Once the spatial modeling process was completed, the object tracking algorithm successfully associated two clusters in the first frame with the same clusters in the second frame. Since the volume centroid value of the box cluster was the same in both frames, the box was identified as a static object. Yet, the volume centroid value of the moving object changed; the horizontal value changed from 65cm to 55cm, indicating that the object moved 10cm in sequential frames. By considering processed time, the tracking algorithm proved that the moving object was moving horizontally at 21.6cm/sec from the right to the left. Finally, the path planning algorithm was able to generate a collision-free path with the shortest travel distance. The algorithm determined that the moving object was moving quickly enough without interrupting the trajectory of the mobile robot. Thus, no new nodes were added on the shortest path calculation, and the robot was able to directly travel from the starting position to the ending position without any disturbance (Fig. 4).

(a) Snapshot from the experiment

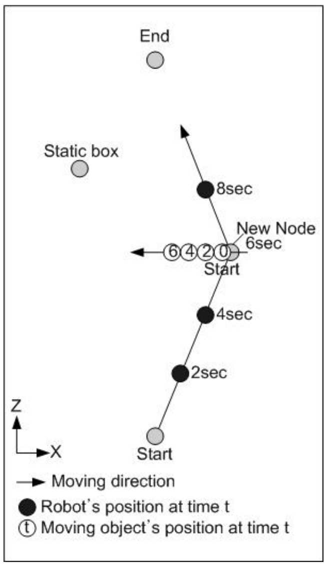


(b) 2D plan view of object positions at time t

FIG. 4: Mobile robot's movement without interruption.

In the second experiment, the mobile robot moved at 40cm/sec, a faster velocity than that of the robot in the previous experiment. Conversely, the moving object in the second experiment moved more slowly than its counterpart in the first experiment. The volume centroid value of the box cluster was set at (-90, 40, 320cm) in both frames. Thus, the box was classified as a static object. On the other hand, the volume centroid value of the moving object varied between frames; in the first frame it was (75, 50, 215cm) and in the second frame it was (70, 50, 215cm). Based on this position difference, the tracking algorithm determined that the CMU block was moving at 10.7cm/sec from the right to the left. While the robot's path was not interrupted by the moving object in the first experiment, it was interrupted by the moving object in the second experiment. This required the algorithm to add a new node (87, 0, 215cm) to prevent a possible collision. The new node created the shortest motion trajectory, a path that was a right-sided route from the start to the end. The total travel distance was 482.6cm. Fig. 5 shows the movement of the mobile robot following the calculated path.

(a) Snapshot from the experiment



(b) 2D plan view of object positions at time t

FIG. 5: Mobile robot's movement with interruption.

4.1.2 Crash avoidance in a 3D open environment

After the basic experiment was completed, a more sophisticated experiment was implemented for verifying whether the proposed crash avoidance framework was able to distinguish 3D open space. To simulate this kind of 3D open space, the research team constructed a bridge for the experiment (Fig. 6). The starting position of the mobile robot was set at (0, 0, 0cm) and the ending position was set at (0, 0, 450cm). As expected, the spatial modeling algorithm was able to detect a space under the bridge as 3D open space. The tracking algorithm identified the bridge as a static object with a volume centroid value (-10, 140, 225cm). Because the bridge did not affect the mobile robot's motion, the path planning algorithm identified the direct path to the target position as the shortest path (Fig. 7).

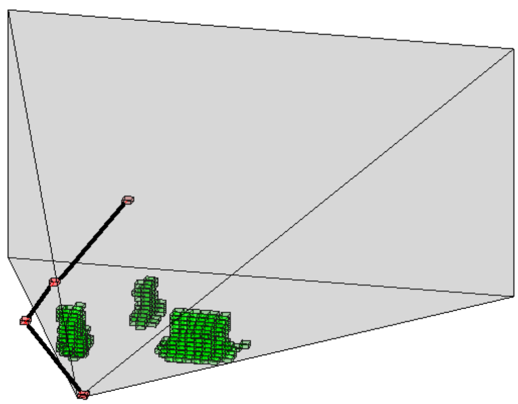
FIG.6: Bridge.

FIG. 7: Mobile robot's movement in the 3D open environment.

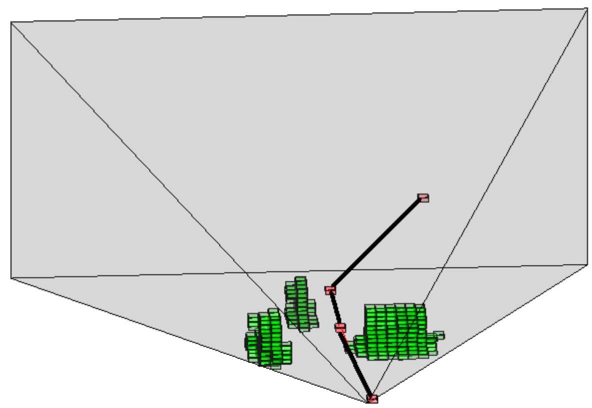
4.2 Crash avoidance for manipulator operation

Unlike most mobile heavy equipment, manipulators moves in three dimensions. Though the mobile robot was an effective research tool for the 2D experiments, no testbed was available for the 3D experiments on manipulator operation. Thus, instead of conducting a laboratory experiment, the research team created a computer simulation model. The simulation environment was constructed in the FSCAL and included two static pipes and one static box (Fig. 8(a) and (b)). The SR-2 mounted on a tripod scanned images from the environment, and the path planning algorithm produced a safe motion plan based on this chaptered information. The processed results were manually exported into Matlab software for visualization.

The manipulator's starting position was set at (0, 125, 0cm), the same setting of the sensor location. Two different ending positions were set in the 3D space in order to allow the path planning algorithm to generate different motion routes based on goal positions. In the first simulation, the target position was set at (-100, 150, 500cm). The path planning algorithm determined that the shortest motion trajectory had a travel distance of 538cm. This path started from (0, 125, 0cm), stopped at (-120, 130, 180cm) and at (-120, 130, 290cm), and finally reached (-100, 150, 500cm). Fig. 8(b) shows the motion route through the plotted environment. In the second simulation, the goal position was set at (100, 150, 500cm). The travel distance was again 538cm, but the route was different. The path began from the same location and went through two new positions (-10, 100, 220cm), and (-10, 100, 320cm). These were points on the straight line between the pipe and the box. The path finally terminated at the goal position. Fig. 8(c) shows this motion path within the modeled environment.



(a) Path to the left-sided goal



(b) Path to the right-sided goal

FIG. 8: Manipulator motion planning.

5. CONCLUSIONS

A preliminary crash avoidance framework for equipment control systems was presented in this paper. It is composed of four major components: data acquisition, spatial modeling, object tracking, and path planning. The experimental results successfully satisfied the proposed hypotheses. All experiments from data acquisition to path calculation were automatically processed in real-time, which validated the first hypothesis. Static and moving objects were detected and tracked as the second hypothesis expected. Lastly, safe motion trajectory was generated while satisfying the third hypothesis. In sum, the proposed crash avoidance framework detected objects from the local environment, tracked their moving trajectories, and finally produced collision-free motion route in real-time.

Despite these encouraging preliminary results, more studies are necessary for further verification and validation of the proposed framework. The proposed framework highly depends on the accuracy of the clustering results. The object tracking algorithm might fail when objects in the scanned image are not correctly segmented. Thus, additional research on clustering algorithms is recommended. In addition to the controlled laboratory settings, more realistic experiments on complex and dynamic construction sites need to be conducted. The proposed framework must also consider specifications of heavy construction equipment such as volume, required working space, degree of freedom, and motion constraints. Next, the optimum number and position of sensors mounted on heavy equipment needs to be investigated. This can be defined based on sensor coverage, specific equipment operation needs, and economic feasibility analysis. With further research, the proposed framework can potentially improve safety for heavy equipment operation by decreasing equipment-related fatalities and property damage.

6. ACKNOWLEDGEMENTS

This study is being supported by the National Science Foundation under Grant Number CMS 0409326. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

7. REFERENCES

Abderrahim M., Garcia E., Diez R. and Balaguer C. (2005). A mechatronics security system for the construction site, *Automation in Construction*, Vol. 14, No. 4, 460-466.

Ali M.S.A.D, Babu N.R. and Varghese K. (2005). Collision free path planning of cooperative crane manipulators using genetic algorithm, *Journal of Computing in Civil Engineering*, Vol. 19, No. 2, 182-193.

Bureau of Labor Statistics. (2005). *Census of fatal occupational injuries*, U.S. Department of Labor, Washington, DC.

Business Roundtable. (1997). *Confronting the skilled construction work force shortage: a blue print for the future*, New York.

Chambon S. and Crouzil A. (2004). Towards correlation-based matching algorithms that are robust near occlusions, *Proceedings of 17th International Conference on Pattern Recognition*, Cambridge, UK.

Cochran S.D. and Medioni G. (1992). 3-D surface description from binocular stereo, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 10, 981-994.

Dellaert F., Thrun S. and Thorpe C. (1998). Jacobian images of super-resolved texture maps for model-based motion estimation and tracking, *Proceedings of 4th Workshop on Applications of Computer Vision*, Princeton, NJ.

Dowman I.J. (1998). Automating image registration and absolute orientation: solutions and problems, *Photogrammetric Record*, Vol. 16, No. 91, 5-18.

Gavrila D.M. and Davis L.S. (1996). 3-D model-based tracking of human in action: a multi-view approach, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA.

Huttenlocher D.P., Klanderman G.A. and Rucklidge W.J. (1993). Comparing images using the Hausdorff distance, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 9, 850-863.

Jung S.K. and Wohn K.Y. (1998). A model-based 3D tracking of rigid objects from a sequence of multiple perspective views, *Pattern Recognition Letters*, Vol. 19, No. 6, 499-512.

Kanade T. and Okutomi M. (1994). A stereo matching algorithm with an adaptive window: theory and experiment, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 9, 920-932.

Kim C., Haas C.T., Liapi K.A. and Caldas C.H. (2006). Human-assisted obstacle avoidance system using 3D workspace modeling for construction equipment operation, *Journal of Computing in Civil Engineering*, Vol. 20, No. 3, 177-186.

Kim C. (2004). Spatial information acquisition and its use for infrastructure operation and maintenance, *Ph.D. Dissertation*, Department of Civil Engineering, University of Texas at Austin.

Kim S., Russell J.S. and Koo K. (2003). Construction robot path-planning for earthwork operations, *Journal of Computing in Civil Engineering*, Vol. 17, No. 2, 97-104.

Koller D., Daniilidis K. and Nagel H.H. (1993). Model-based object tracking in monocular image sequences of road traffic scenes, *International Journal of Computer Vision*, Vol. 10, No. 3, 257-281.

Koo H. and Jeong C. (2001). An area-based stereo matching using adaptive search range and window size, *Proceedings of the International Conference on Computational Science, Lecture Notes in Computer Science*, 2074, 44-53.

Lee S. and Adams T.M. (2004). Spatial model for path planning of multiple mobile construction robots, *Computer-Aided Civil and Infrastructure Engineering*, Vol. 19, No. 4, 231-245.

Marapane S. and Trivedi M.M. (1994). Multi-Primitive Hierarchical (MPH) stereo analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, No. 3, 227-240.

Marchand E., Bouthemy P. and Chaumette F. (2001). A 2D-3D model-based approach to real-time visual tracking, *Image and Vision Computing*, Vol. 19, No. 13, 941-955.

Miller M.K., Winkless N. and Bosworth J. (1998). *The Personal Robot Navigator*, Robot Press, Conifer, CO.

Moravec H. and Elfes A. (1985). High-resolution maps from wide-angle sonar, *Proceedings of IEEE International Conference on Robotics and Automation*, Washington, DC, 116-121.

Nickels K. and Hutchinson S. (2001). Model-based tracking of complex articulated objects, *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 1, 28-26.

Oloufa A.A., Ikeda M. and Oda H. (2002). GPS-based wireless collision detection of construction equipment, *Proceedings of the 19th International Symposium on Automation and Robotics in Construction (ISARC)*, National Institute of Standards and Technology, Gaithersburg, Maryland. 461-466.

Olson C. (1998). A probabilistic formulation for Hausdorff matching, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Santa Barbara, CA. 150-156.

Polat E., Yeasin M. and Sharma R. (2003). A 2D/3D model-based object tracking framework, *Pattern Recognition*, Vol. 36, No. 9, 2127-2141.

Reddy H.R. and Varghese K. (2002). Automated path planning for mobile crane lifts, *Computer-Aided Civil and Infrastructure Engineering*, Vol. 17, No. 6, 439-448.

Rehg J.M. and Kanade T. (1995). Model-based tracking of self-occluding articulated objects, *Proceedings of 5th International Conference on Computer Vision*, Massachusetts Institute of Technology, Cambridge, MA.

Riaz Z., Edwards D.J. and Thorpe A. (2006). SightSafety: A hibrid information and communication technology system for reducing vehicle/pedestrian collisions, *Automation in Construction*, Vol. 15, No. 6, 719-728.

Ruff T.M. (2000). Test results of collision warning systems for surface mining dump trucks, *Report of Investigations: 9652, Pittsburgh Research Laboratory, National Institute for Occupational Safety and Health*, U.S. Department of Health and Human Services, Pittsburgh, PA.

Rziza M., Tamtaoui A., Morin L. and Aboutajdine D. (2000). Estimation and segmentation of a dense disparity map for 3D reconstruction, *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Istanbul, Turkey.

Saeedi P., Lawrence P.D., Lowe D.G., Jacobsen P., Kusalovic D., Ardron K. and Sorensen P.H. (2005). An autonomous excavator with vision-based track-slippage control, *IEEE Transactions on Control System Technology*, Vol. 13, No. 1, 67-84.

Sarata S., Koyachi N., Tubouchi T., Osumi H., Kurisu M. and Sugawara K. (2006). Development of autonomous system for loading operation by wheel loader, *Proceedings of 23rd International Symposium on Automation and Robotics in Construction*, 466-471.

Schiffbauer W.H. and Mowrey G.L. (2001). An environmentally robust proximity warning system for hazardous areas, *Proceedings of the ISA Emerging Technologies Conference*, Triangle Park, NC. Vol. 2091, p10.

Schwarz C. and Lobo N.V. (2005). Segment-based hand pose estimation, *Proceedings of 2nd Canadian Conference on Computer and Robot Vision*, Victoria, British Colombia, Canada.

Soltani A.R. and Fernando T. (2004). A fuzzy based multi-objective path planning of construction sites, *Automation in Construction*, Vol. 13, No. 6, 717-734.

Soltani A.R., Tawfik H., Goulermas J.Y. and Fernando T. (2002). Path planning in construction sites: performance evaluation of the Dijkstra, A*, and GA search algorithms, *Advanced Engineering Informatics*, Vol. 16, No. 4, 291-303.

Stentz A., Bares J., Singh S. and Rowe P. (1999). A robotic excavator for autonomous truck loading, *Autonomous Robots*, 7, 175-186.

Tan P., Steinbach M. and Kumar V. (2005). *Introduction to Data Mining*, Pearson Education, Inc., Boston, MA.

Teizer J. and Castro-Lacouture D. (2007). Combined ultra-wideband positioning and range imaging sensing for productivity and safety monitoring in building construction, *Proceedings of the 2007 ASCE International Workshop on Computing in Civil Engineering*, Pittsburgh, PA. p681-688.

Teizer J. (2006). *Real-time spatial modeling to detect and track resources on construction sites*, Ph.D. Dissertation, Department of Civil Engineering, University of Texas at Austin.

Tserng H.P., Ran B. and Russell J.S. (2000). Interactive path planning for multi-equipment landfill operations, *Automation in Construction*, Vol. 10, No. 1, 155-168.

Wan T.R., Chen H. and Earnshaw R.A. (2005). A motion constrained dynamic path planning algorithm for multi-agent simulations, *Proceedings of 13th International Conference in Central Europe on Computer Graphics*, Plzen, Czech Republic.

Wei G., Brauer W. and Hirzinger G. (1998). Intensity- and gradient-based stereo matching using hierarchical Gaussian basis functions, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, 1143-1160.

Williams C., Cho Y.K. and Youn J.H. (2007). Wireless sensor-driven intelligent navigation method for mobile robot applications in construction, *Proceedings of the 2007 ASCE International Workshop on Computing in Civil Engineering*, Pittsburgh, PA. p640-647.

Zhang Z. and Faugeras O. (1992). *3D Dynamic Scene Analysis*, Springer-Verlag, Berlin.

Zhijia Z., Shabai H. and Zelin S. (2003). A fast strategy for image matching using Hausdorff distance, *Proceedings of the IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, Changsha, China.

Zhou J. and Shi J. (2002). A robust algorithms for feature point matching, *Computers and Graphics*, Vol. 26, No. 3, 429-436.