Human Assisted Fitting and Matching Primitive Objects to Sparse Point Clouds for Rapid Workspace Modeling in Construction Automation

-건설현장에서의 시공 자동화를 위한 Laser Sensor기반의 Workspace Modeling 방법에 관한 연구-

권 순 욱*
KWON, SOON-WOOK

Abstract

Current methods for construction site modeling employ large, expensive laser range scanners that produce dense range point clouds of a scene from different perspectives. Days of skilled interpretation and of automatic segmentation may be required to convert the clouds to a finished CAD model. The dynamic nature of the construction environment requires that a real-time local area modeling system be capable of handling a rapidly changing and uncertain work environment. However, in practice, large, simple, and reasonably accurate embodying volumes are adequate feedback to an operator who, for instance, is attempting to place materials in the midst of obstacles with an occluded view. For real-time obstacle avoidance and automated equipment control functions, such volumes also facilitate computational tractability. In this research, a human operator' s ability to quickly evaluate and associate objects in a scene is exploited. The operator directs a laser range finder mounted on a pan and tilt unit to collect range points on objects throughout the workspace. These groups of points form sparse range point clouds. These sparse clouds are then used to create geometric primitives for visualization and modeling purposes. Experimental results indicate that these models can be created rapidly and with sufficient accuracy for automated obstacle avoidance and equipment control functions.

Keywords : Fitting and Matching Objects, Sparse Point Clouds, Workspace Modeling, Automation

1. Introduction

The objective of a computer vision system is to recognize visual information to assist the vision system in understanding an automation task (Arman et al. 1990, Arman et al. 1993). In several applications, such as path planning of autonomous equipment, assembly and inspections for manufacturing, and tele-operated control of heavy equipment, three-dimensional (3D) workspace information can be used to update workspace information, prove graphical aid, and provide a mathematical model to compute location and orientation (Johnson et al. 1999). Graphical workspace modeling can bring about improvements in safety while at the same time lessening the need for skilled workers to operate heavy equipment under a wide range of working conditions. There are two general modes in which graphical workspace modeling can be applied for equipment operations: (1) as interactive visual feedback while a piece of heavy equipment is being operated, or (2) as a tool for 3D graphical simulation. In the latter case, application of such a modeling technique can ultimately contribute to an equipment operator's sense of whether-and how-he/she should move before actually proceeding to do so (Kim et al. 000, McLaughlin et al. 2002). In this paper, we address the need for three-dimensional geometrical workspace information for equipment control functions and obstacle avoidance in construction. We present algorithms that accurately fit and match objects with regard to location and orientation to sparse point clouds that have less than 50 scanned points for each object in a construction scene. The implementation of these algorithms will allow a human operator to rapidly construct a world model from unfiltered real-world range data.

* 정회원, 한국건설기술연구원 선임 연구원, 공학박사

1.1 A Range Data

Considerable effort has been devoted to the development of various methods for extracting geometrical information from scenes. This extraction is still a major concern for both computer vision and robot vision (Lebegue et al. 1994, Tsukiyama 1997).

The need for real-time determination of the 3D pose of rigid objects has been pointed out by a number of researchers. Such a system needs to provide full 3D pose estimation of arbitrarily shaped objects in real-time. Such systems have been difficult to develop for the three reasons: (1) there is less sensitivity for accuracy of 3D pose estimation in two dimensional video image data, (2) the identification of features and correspondence with the objects in the database are difficult problems in current 3D pose estimation applications, and (3) real-time 3D pose estimation is difficult to achieve due to the high computational load of existing methods (Simon et al. 1993).

Laser range scanner has been widely used to obtain 3D range data for construction site scenes (Cheok et al. 1999). The uses of laser range scanners in the construction industry include: (1) generating 3D as-built model, (2) tracking terrain changes due to excavation, (3) site inspection, (4) aerial surveying, and (5) process simulation and operator training. Unlike traditional survey methods, range data can be captured rapidly because no target reflectors are required. Unlike ultrasonic and stereo vision, the laser range scanner provides a large amount of precise data.

Research related to the use of laser range scanner in construction has focused on issues such as capturing 3D-digital conditions of the construction project, tracking construction process in order to enable project control, and integrating construction process information into life cycle data. Laser range scanners have proven to be beneficial for tele-operable control of semi-automated or automated equipment on large construction sites where timely, on-site decisions require rapid recognition and accurate measurement of objects in the workspace.

A limitation of most workspace modeling applications is their reliance upon analyzing dense point cloud data, which requires computationally intensive processing that cannot be used for real-time equipment control. The low accuracy in extracting objects from dense point clouds is an additional limitation of current modeling systems. For real-world robotic, semi-automated, or automated equipment, obtain geometric information of target objects must be obtained rapidly (Lebegue et al. 1994, Vemuri t al. 1986).

Current workspace modeling applications demand large amounts of computation due to very large data sets. Furthermore, these algorithms may not be appropriate for symmetric objects (Johnson 1997) since their representation of these kinds of objects makes it difficult to execute automated path planning due to the high computational cost of scanning each point on the surface (Sabata t al. 1996).

1.2 Usage of range information in construction automation

Construction site environmental characteristics vary in temperature, humidity, sound, lighting, and air quality. Noise from equipment and other disturbances can cause miscommunication between an equipment operator and the person who directs the operator. The physical environment (i.e. a limited view from the operator's position, depth perception limitations, and miscommunications) of the construction site can cause collisions between structures and equipment such as truck-mounted concrete pumps, man-lifts, backhoes, or manipulators (Cho et al. 2002). Furthermore, changes are continuously introduced in building units, target materials, and equipment positions, and the operator has to continuously set up-to-the-minute relative positions in the conventional working environment. Repetitive equipment set up is time consuming and lowers productivity.

Results from the Occupational Safety and Health Administration (OSHA) study on construction fatalities in 1985-1989 reveals that construction accidents can be categorized into five types: (1) falls from elevation (33%), (2) struck-by accidents (22%), (3) cause-in/between incidents (18%), (4) electrical shock (17%), and (5) all other factors (10%) (Haas et al. 1995). In many cases, construction equipment operations have played role in such fatalities.

To improve safety and address the limitations of the work environment, more accurate equipment controls are needed. Current research shows that graphical models that help visualize geometric information using descriptive three-dimensional models can improve equipment controls for construction operations such as material handling, heavy lifting, and earth moving (Kim et al. 2000, Stenz et al. 1998, Cheok et al. 2000, Cho et al. 2001, Cho et al. 2002).

The use of range scanners for obtaining 3D range data for a construction site scene is increasing quickly (Cheok et al. 2000). It is possible to rapidly capture range data because, unlike traditional surveys, no target reflectors are required. Nevertheless, rich and raw data provided by range scanners and cameras are not sufficient for automating the manipulation operation. If there is no application to manipulate scanned data, these raw data cannot provide site

information directly. Workspace modeling, the construction of a virtual representation of the equipment's working environment, can be employed to facilitate automation tasks (Johnson et al. 1997).

1.3 3D object recognition methods using range data

To date, most of the research studies using range data have concentrated on developing a 3D object recognition method. Object recognition is a matching process between scene and model description. Efficient matching is complicated by the variability of object types, the difficulty in modeling some types of objects, and the need to efficiently verify of the accuracy of the fully automated equipment operation.

Johnson and Herbert (1999) developed an object recognition system using spin images, which are used for matching different surfaces for efficient object recognition in cluttered 3D scenes. They used surface matching for model-based object recognition as well as aligning two surfaces of the same object represented in two different coordinate systems for the purpose of obtaining a transformation between two coordinate systems.

Their approach matches surfaces based on matching individual points on two different scanned surfaces. The two surfaces are matched when the images of many points on the surfaces are similar. It is difficult to define surface points so that they can be differentiated from one another. Even though these researchers developed a new method that can recognize objects by collating object information stored in sensed 3D points, their method was not very efficient because it required a large amount of time to filter dense point clouds.

Johnson and Hoffman (1997) created a system for the semi-automatic modeling of complex environments. Their system, called Artisan, consists of laser range sensors and object modeling applications. Artisan uses various kinds of 3D sensors, such as laser range finders and stereo systems, to acquire image and range data. Artisan also recognizes and locates objects using a matching algorithm that can fuse the data taken from multisensor viewpoints. This application has several advantages for workspace modeling; however, it still has limitations such as difficulties in recognizing shiny metal objects, and cylindrical object, represented object models only on a single scale, and requiring a heavy computational load for merging images and filtering data

Sabata and Aggarwal (1996) used a novel procedure to find surface correspondence based on the hypergraph representation. Using planar and quadric surface pairings, motion transformation was computed. Segmentation was performed in two steps. First, the over-segmentation method using zero and first-order local surface properties was performed, followed by the use of high order surface representations to segment dense range images.

1.4. Use of 3D laser scanning in construction industry

The Autonomous Loading System (ALS) was developed at the Robotics Institute at Carnegie Mellon University (Stenz et al. 1998). ALS is a fully automated truck-loading excavator. This system uses two laser range scanners whose purpose is to recognize and localize the truck, detect obstacles, and measure the soil face. It is assumed that the hauling truck is parked on one side of the excavator. The segmented data can be matched region by region to the simple model of the truck bed. The identified four corner points of the sensor data are used to compute the position and orientation of the truck bed.

One of the disadvantages in ALS is that whenever a new truck arrives for loading, the ALS must repeat the computing process of matching model-to-scene, even though all of the trucks are identical in shape. This results in decreased efficiency because the most important issue of ALS is to ascertain object location, not to recognize the object. Since human operator knows the object in advance, collaboration between humans and ALS would be valuable at this point: human have good decision making ability despite incomplete or redundant information, while ALS provides computer control skills when executing repetitive motion control.

Cyra Corporation has developed a laser range scanning system (Cyrax) that extracts 3D data points of the work environment and involves a semi-manual assisted 3D model regeneration method using point clouds (2002). Cyrax combines a high-resolution distance measurement sensor with software that creates 2D drawings and 3D models that are exportable for industry standard CAD and graphical modeling (IGES, AutoCAD DXF, Microstation DGN, ASCII, BMP, and JPEG). Although the scanning process of this system provides more precise as-built 3D models relatively faster than other traditional manual measurement and design systems, it still requires days or weeks for the modeling process, which is not useful for real-time decision making in construction operations.

NIST developed a system that can measure terrain changes due to excavation at a construction site and demonstrate the results in a real-time terrain assessment (Cheok et al. 2000, Stone et al. 2000). A Riegel laser range scanner was used to scan terrain. The scanned

terrain data was derived from two locations around the construction site at the end of the workday. Post-processing consists of four steps: (1) registering of the two scans based on the pre-defined positions of the scanner, (2) fine tuning the registration visually to decide the regions, (3) selecting of the region of interest using the scanned data, and (4) computing a volume. Some of the disadvantages observed in field-testing were that the battery used for the power supply needed to be recharged after each use; the second scanner mounted on the pole in the center of the construction site was disturbed by dirt piles and construction equipment located in front of the scanner; manual fine-tuning of the scanned data to perform merging was extremely time-consuming; and computing supplementary volume was awkward after the erection of the structure began.

2. The limitations of current methods and discussion of the proposed methods

Most current methods for modeling work environments rely on analyzing dense point cloud data, which require computationally intensive processing and usually require longer periods of time than the ongoing construction operation. The low accuracy in extracting objects from dense clouds using large and expensive scanners ($30k-$100k) is an additional limitation of full area range scanning and fully automated object recognition methods. Cho (2001) has shown that by exploiting a human operator's ability to recognize objects in a construction scene, pre-stored graphic representations of construction objects can be matched and fitted to measured data in the construction environment.

Most of the related research studies recently completed have focused on automated object recognition to create world models (Ebest et al. 2000), and very little research has examined the use of a human operator's ability to recognize objects in a construction site work environment. The human ability to recognize objects is very valuable for rapid workspace modeling of construction site scenes since the object recognition algorithms currently being used require heavy computational loads and present accuracy problems (Kim et al. 2000, Cho et al. 2001)

Since most objects in a construction site are known and manmade, they can be graphically generated and stored in an object database as parametrically defined object classes. By exploiting a human operator's ability to recognize such objects, objects can be matched and fitted to sensed data from 3D position sensors.

The ability to extract models of real world objects in a construction workspace for equipment operations from only a limited number of scanned points is a significant advantage of this approach over full range scanning methods that require more intensive computational loads.

This paper presents algorithms that accurately fit and match objects, with regard to location and orientation, to sparse point clouds which have less than 50 scanned points for each object. The implementation of the algorithms will allow a human operator to rapidly construct a world model from unfiltered real-world range data.

With respect to the geometric primitives most frequently encountered in a construction site, it appears that a few types of objects can be used to model a wide range of construction scenes. Planar surfaces can be used for the partition of workspaces such as fences and walls. Cuboids can be used for fitting and matching structural objects such as columns, box-beams and walls, and for finishing objects. Cylinders can be used to fit and match chemical pipes, ventilation pipes, and concrete piles. The following fitting and matching algorithms were developed for these primitives:

(1) Planar surface fitting and matching algorithms (included in cuboid algorithm)
(2) Cuboid fitting and matching algorithms
(3) Cylindrical object fitting and matching algorithms
(4) Sphere fitting and matching algorithms

3. Experimental setup and human-assisted object fitting and matching process

A single-axis laser range finder, a pan/tilt unit (PTU), and a personal computer were used for the experimental set up.

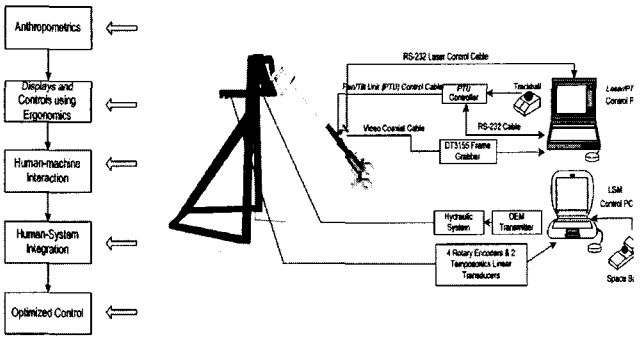

Figure 1. Laser Scanning System and Computer Controlled Large Scale Manipulator

The single-axis laser range finder (DistoMemo) that is mounted on the PTU is designed not only for hand-held operation, but also for

computer use through an interface.

The measurements can be remotely taken and transferred directly into the computer. The range of measurement of the laser range finder is 100 m with an accuracy of ± 3 mm . The step size of the tele-operated PTU, which controls the laser range finder, is of high resolution (0.0128571° /step) and its maximum speed is a little over 60° /second. Its error is 0.2 cm for every 10 min motion. Figure 1 shows an experimental setup that consists of a computer controlled large scale manipulator (LSM) and laser scanning system at The University of Texas at Austin's Field Systems and Construction Automation Lab (FSCAL). It can be used for pre-programmed automated control experimentation after finishing workspace modeling.

The sparse points cloud is acquired by the operator picking points for each object using a single-axis laser range finder. The modeling process involves the following functions:

(1) Select an object for scanning (by operator)
(2) Acquire sparse point cloud data in the form of range images
(3) Convert range data into xyz coordinates
(4) Analyze the features of each surface of the object
(5) Match all of the object surfaces with the model's surfaces using matching algorithms
(6) Fit the object into the point cloud using a fitting algorithm

4. Object fitting and matching algorithms

Graphical workspace modeling can improve construction equipment control and operations. Equipment operators can use graphical workspace models as an interactive visual feedback tool while controlling equipment (Kim et al. 2000, McLaughlin et al. 2002). For the rapid modeling of construction site objects from sparse point clouds three basic algorithms have been developed that address construction site objects. These are: (1) the cuboid fitting and matching algorithm, (2) the solid and hollow pipe fitting and matching algorithm,(3) the sphere algorithm, and (4) the planar algorithm. Since cuboid and cylinder types of primitives consist of 6 planar surfaces (cuboid), and two planar surfaces and one curved surface (cylinder), the algorithms were developed as a surface based fitting and matching method. Algorithm development and revisions were based on lab experiments. By using these algorithms we achieve: (1) accurate and reliable methods to save computational cost and time, (2) improved fitting and matching methods to attain real-time execution, and (3) increased modeling accuracy with operator's assistance. Figure 2 shows the entire fitting and matching process.

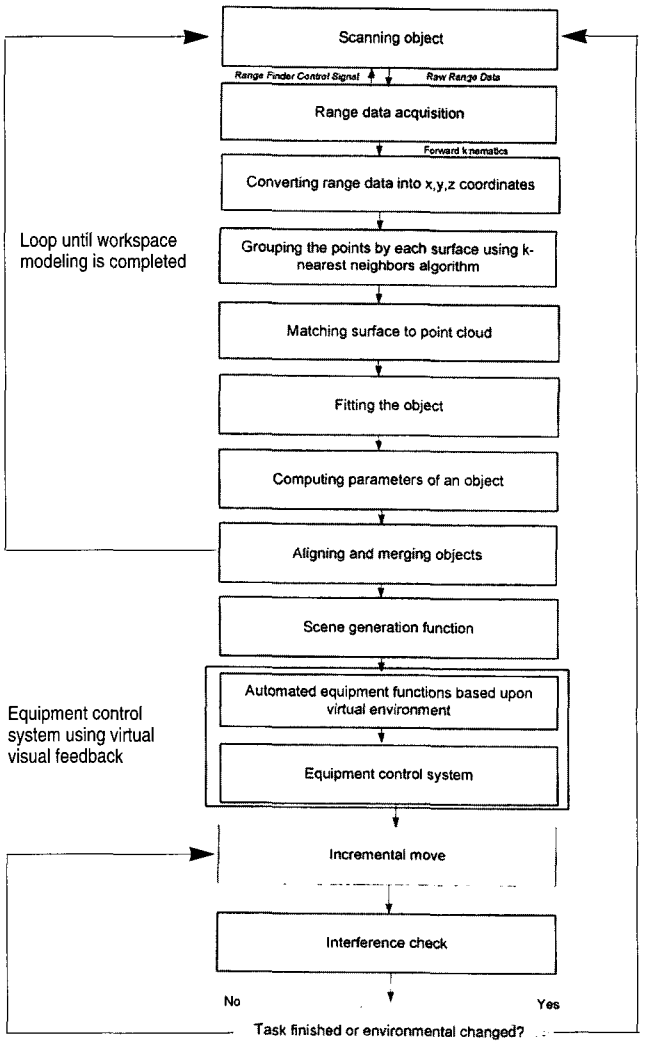

Figure 2. Fitting and Matching Process

4.1 Cuboid fitting and matching algorithm

A bounded cuboid is described by a set of vertex points vp= {a, b, c, d, e, f, g, h}, and is composed of 6 surfaces. A bounded plane, one of the cuboid's six surfaces, is represented by a set of parameters p = {p1, p2, p3, p4} that defines a plane, and a set of edge points E that lie in the plane and describes the vertices of the plane's boundary. The cuboid algorithm is used to find plane parameters for surfaces such as normals of all planes and vertex points. An estimate of the plane equation is found by the k-nearest neighbors algorithm. Then the optimized plane is defined by the least squares method. This method computes the best-fitted plane formed from a set of scanned points u = {(Xi, Yi, Zi)}, which are scanned by an operator on the

viewable three surfaces of a cuboid. After iterating the above steps on the other surfaces, the intersecting edges and vertices are found from the estimated three plane equation parameters. Consequently, the parameters of the cuboid are computed using the edges and points on each surface of the cuboid.

The cuboid algorithm consists of four steps:

(1) K-nearest neighbors to segment points onto each surface of the cuboid
(2) Plane optimization using the least squares method to fit surfaces of the cuboid
(3) Find the intersecting edge line between two surfaces and the vertices
(4) Points projection and parameters computations

1) *K*-nearest neighbors:To find the nearest points from all measured points in a cuboid, the k-nearest neighbor algorithm was developed. This algorithm finds the nearest two points in a 3D space by computing all distances from one scanned point to other scanned points. After finding the two nearest neighbor points of each scanned point, we can get the list of all three-point sets. Then the normal vector for each set of three-point sets can be computed. Using these normals, the scanned points can be segmented by each cuboid surface. The algorithm developed uses the following procedure:

(1) Compute the distance from each single scanned point to the other scanned points.
(2) Sort the pair-wise distance measurement data to determine the two nearest neighbors to each single point, thereby forming a three-point set for each point.
(3) Make an equation of a plane that contains the three-point sets associated with each single point.
(4) Compute the normal vector of each plane.
(5)Sort all normals and determine the three most frequent normals that represent three surfaces of a cuboid.
(6) Segment all the points into the three surfaces by assigning each point to the closest surface.
(7) Repeat (loop) the above steps for every scanned point.

2) Plane optimization using the least squares fitting method: To find the optimized plane using the segmented points on each surface, the least squares fitting method was applied. In a linear equation the predicted Z or "(1)", the error term is "(2)". Given a set of data points (xi, yi, zi), determine the values of *A*, *B*, and *C* so that the predicted plane ẑ minimizes the sum of the squared residuals, "(2)". This function is nonnegative and its graph is a hyperparaboloid whose vertex occurs when the gradient satistfies $\nabla G = (0, 0, 0)$. This leads to a system of three linear equations in which *A*, *B*, and *C* can be easily solved.

$$\hat{Z}_i = A_{x_i} + B_{y_i} + C \qquad (1)$$

$$\sum_{i=1}^{n} (Z_i - \hat{Z}_i)^2 \qquad (2)$$

$$(0,0,0) = \nabla G = 2\sum_{i=1}^{m} [(Ax_i + By_i + C) - z_i](x_i, y_i, z_i) \qquad (3)$$

$$Q = \begin{bmatrix} \sum_{i=1}^{m} x_i^2 & \sum_{i=1}^{m} x_i y_i & \sum_{i=1}^{m} x_i \\ \sum_{i=1}^{m} x_i y_i & \sum_{i=1}^{m} y_i^2 & \sum_{i=1}^{m} y_i \\ \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} y_i & \sum_{i=1}^{m} m \end{bmatrix} \quad n = \begin{bmatrix} A \\ B \\ C \end{bmatrix} \quad d = \begin{bmatrix} \sum_{i=1}^{m} x_i z_i \\ \sum_{i=1}^{m} y_i z_i \\ \sum_{i=1}^{m} z_i \end{bmatrix} \qquad (4)$$

$$n = Q^{-1} d \qquad (5)$$

3) Finding the intersecting edge line between two surfaces and the vertex (matching point of cuboid): The line of intersection of two planes can be found by solving the two linear equations representing the planes. After appling these equations to all three surfaces, we can find three intersection edges. Therefore, a vertex of a cuboid can be obtained from those three edges.

Point projection and computation of parameters of the cuboid: After segmenting the points into the three surfaces of the cuboid, the points should be projected onto the optimized surface. It is assumed that two points, *(x1, y1, z1)*, *(x2, y2, z2)*, are selected from the optimized plane. The size of the cuboid can be determined by computing the distance from each edge to the farthest point on a certain surface. The distance *d* from point *K* to a line defined by the end point *P1* and the direction *V* can be found by calculating the magnitude of the component of *K-P1* that is perpendicular to the line as shown in figure 4. The squared distance between the point *K* and the line can be found by subtracting the square of the projection of *K-*

P1 in the direction V from the square of *K-P1*. This provides us:

$$d^2 = (K - P_1)^2 - [proj_D(K - P_1)]^2$$

$$= (K - P_1)^2 - [\frac{(K - P_1) \cdot V}{V^2} V]^2 \qquad (6)$$

$$= (K - P_1)^2 - \frac{[(K - P_1) \cdot V]^2}{V^2}$$

Figure 3 to 6 show the results of the cuboid fitting and matching algorithm.

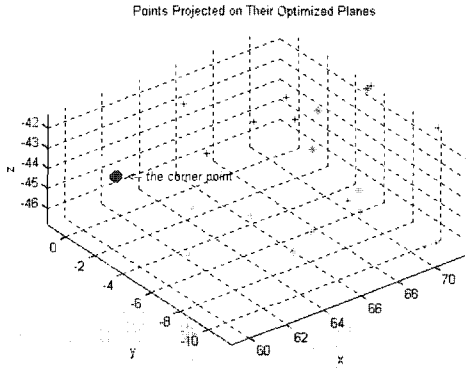

Figure 3. Segmenting the Points

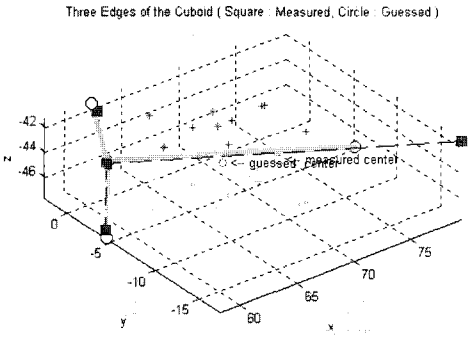

Figure 4. Three Edges of a Cuboid

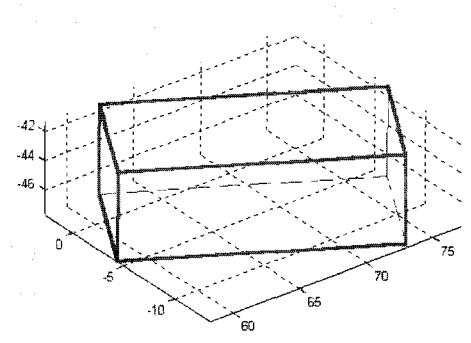

Figure 5. Fitted and Matched Cuboid

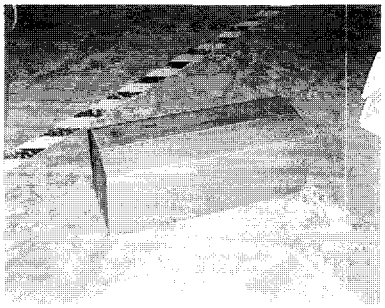

Figure 6. Actual Object

4.2 Solid cylinder fitting and matching algorithm

Four parameters can define a bounded cylinder: *a* scalar radius *r*; an axis vector, *a*; a center point to identify the axis vector, *c* = (*Xc, Yc, Zc*); and a set of scanned points *d* = *{(Xi, Yi, Zi)}* that define the boundary of the cylinder. This algorithm uses the nearest neighbor algorithm to define the normal vector. Four scanned points, chosen with the starting point, are used to find the planar surface of the cylinder. The estimated planar surface is applied to group the points into planar and curved surfaces. By projecting the points on the curved surface, parameters r and c can be estimated. The radius of the circle is found as the distance from the center of the circle to any point on the optimized curve. These projected points are connected into chords in the circle and are used to estimate the center of the circle $\hat{c}$. A primary estimation of the radius, $\check{r}$, is found by $\check{r}$ =mean($\hat{c}$ - k') (k'={the projected points on the optimized curve of planar surface}). Consequently the final values of *a, c, r* are found by the least squares method using data d. Figure 9 shows the segmented points on the planar and curved surface of a solid cylinder. Figure 8 shows the points projected from the curved surface onto the planar surface of the solid cylinder. Figure 9 shows the complete solid cylinder model computed by the scanned points from the actual object shown in Figure 10.

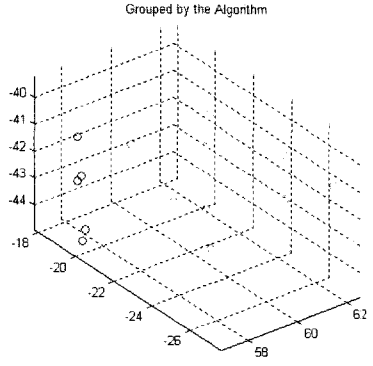

Figure 7. Segmentation (1)

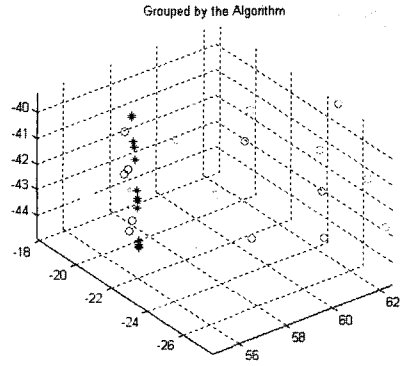

Figure 8. Segmentation (2)

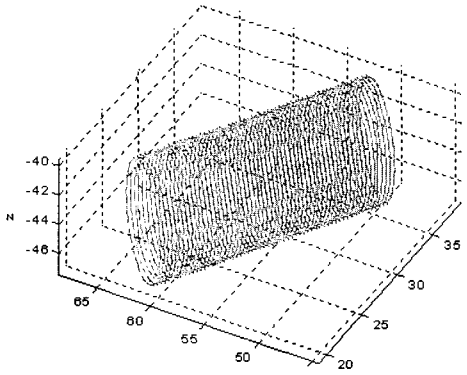

Figure 9. Modeled Solid Cylinder

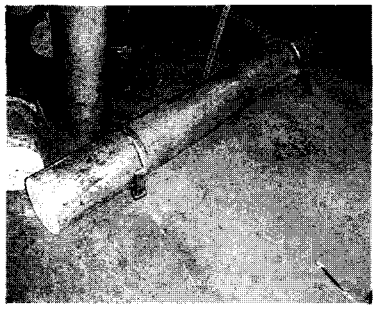

Figure 10. Actual Cylinder

4.3 Hollow cylinder fitting and matching algorithm

In this hollow cylinder fitting and matching method, Principal Components Analysis (PCA) was used to determine the primary axis of cylinder. Excluding the steps for computing the primary axis, the other steps of the algorithm follow the same sequence as the solid cylinder algorithm.

It is important to diminish dimensionality by parsimonious data reduction techniques. PCA is a distribution-based ordination method in which the distance between sites in an ordination diagram is correlated with multi dimensional distribution (Lebegue et al. 1992). PCA assumes that all vectors in a set of *n* dimensional samples *a1*··· an can be explained by a single vector *a0*. The vector *a0* is derived using the least squares method, in which the sum of the squared distances between *a0* and the various *ak* are minimized. We define the square-error criterion function F0(*a0*) by

$$F_0(a_0) = \sum_{b=1}^{n} \|a_0 - a_k\|^2 \tag{5}$$

$$p = \frac{1}{n} \sum_{b=1}^{n} a_b \tag{6}$$

$$F_0 = \sum_{b=1}^{n} \|(a_0 - p) - (a_b - p)\|^2 \tag{7}$$

Projecting the sample data onto a line through the sample mean, one-dimensional representation can be computed. If we let e be a unit vector of the line direction, the line equation is

$$a = p + de \tag{8}$$

Scalar d is the distance between the sample data and the sample mean *p*. We can find the coefficients *dk* by minimizing the squared criterion function.

$$F_0(d_1,...,d_n,e) = \sum_{b=1}^{n} \|(p + d_b e) - a_k\|^2 = \sum_{b=1}^{n} \|d_b e - (a_b - p)\|^2 \tag{9}$$

$$d_b = e^t (a_b - p) \tag{10}$$

The best direction e of the line can be found by solving scatter matrix U, which is defined by

$$U = \sum_{b=1}^{n} (a_b - p)(a_b - p)^t \tag{11}$$

$$F_1(e) = -e^t U e + \sum_{b=1}^{n} \|a_b - p\|^2 \tag{12}$$

LaGrange multipliers can be used to maximize the *etUe*, which is subject to the constraint ||e||=*1*. Let φ be an undetermined multiplier. We can do differentiation of

$$v = e^t U e - \phi(e^t e - 1) \tag{13}$$

with regard to e getting

$$\frac{\partial u}{\partial e} = 2Ue - 2\phi e \tag{14}$$

By setting the gradient vector equal to zero, we see that e should be an eigenvector of the scatter matrix. The eigenvector will be the

primary axis of the hyper-ellipsoid that can be obtained by reducing the dimensionality of the feature space and by restricting attention to the directions along the scatter of the cloud (Vermuri et al. 1986, Schweikert 1966). It will be the primary axis of the cylinder.

$$Ue = \phi e \qquad (15)$$

After finding the primary axis of a cylinder, the estimated planar surfaces can be generated on the top and bottom of a hollow cylinder. By projecting the points of the curved surface onto the planar surfaces, the radius and center point of the hollow cylinder can be estimated. The radius of the circle is found using the same method used in the solid cylinder algorithm. Consequently the final values of the radius, center point, and primary axis are found by this fitting and matching method using scanned data. See Figure 11 to Figure 14.

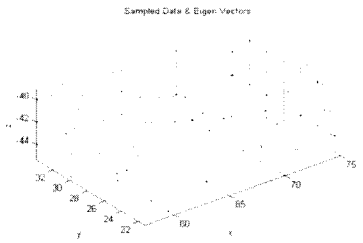

Figure 11. Scanned Points

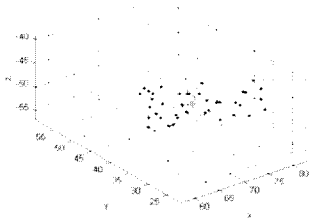

Figure 12. Result of Fitting and Matching(1)

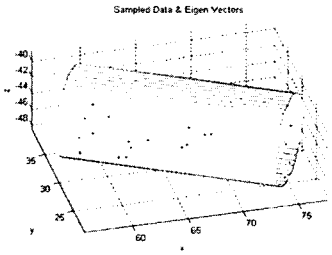

Figure 13. Result of Fitting and Matching (2)

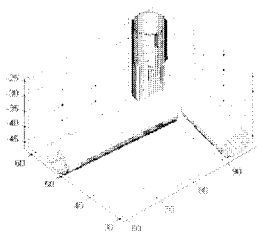

Figure 14. Modeled Hollow Cylinders

4.4 Sphere fitting and matching algorithm

To represent the boundary of a sphere, four points should be measured by a laser range finder. The four points are generated from two adjacent triangles on the surface of a sphere. Euclidian geometry states that the perpendiculars to the plane of each triangle at their centers of gravity pass through the center of sphere (see Figure 15). Then, the radius and exact location in space of the center of the sphere can be easily computed (see Figure 15).

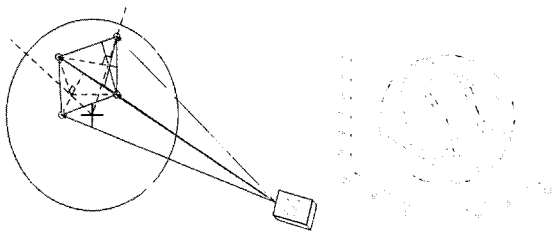

Figure 15. Scanning and Modeling a Sphere Object

4.5 Conclusions and Experimental results

The fitting and matching algorithms discussed in this paper are an integral part of a method that involves several other functions including: human object recognition, collecting range information, segmenting scanned points, and computing parameters resulting in final fitting and matching for construction equipment operation and as-built modeling. A key advantage of the method is that it incorporates human cognitive ability to recognize and classify objects in the workspace, that is, a human operator initiates scanning, recognizes objects, and controls the system for data acquisition. In addition, fitted and matched objects are verified by the operator and then inserted into the workspace model and provides more rapid as-built 3D models relatively faster than other traditional manual measurement and design systems, they still require days or weeks for the modeling process due to their data densities

Experiments were conducted to determine the efficiency of the human assisted modeling method using real material such as 6, 8, 10, 12 inch-diameter pipe and boxes for construction material.. The algorithms, which are based on the least squares method and PCA method, were found to be useful for modeling construction objects of cylindrical and cuboid shapes. They were applied to determine the widths, depths, and heights of cuboids, and the diameters and lengths of solid cylinders, including the locations and orientations.

Experiments were conducted using different numbers of scanned points, various view angles, and various sizes of objects. Table 1 shows experimental results of the cuboid fitting and matching

algorithms. Table 2 shows the results of hollow cylinder algorithm. The algorithms tested yield approximately less than 2-degree angular deviation between the model and the real object's axis. In all tests the size difference between the modeled and the actual object's surfaces was less than 5 %. For increased accuracy, further modifications of the algorithms are required. In general, low deviation values and the low modeling times in Table 1 and 2 indicate that a system based on the above geometric algorithms and a human-guided simple laser range finder can model construction objects rapidly and with sufficient accuracy.

The approach taken in this research relies on a human's cognitive ability to recognize and classify objects in the workspace. Much research has been conducted on automatic object recognition for model generation, but these methods are neither robust nor efficient enough for real-time modeling in construction automation. The goal here is to balance human discernment and efficient range data acquisition with the proper exploitation of the computer in the areas of model generation, interference checking and avoidance control.

ACKNOWLEDGEMENTS

This paper is based on the research funded by the National Science Foundation (Grant#: CMS-0000137) and the National Institute Standard and Technology (Project NBR: NA1341-02-W-0742). The authors gratefully acknowledge their financial support and encouragement throughout this study.

REFERENCES

1. F. Arman, and J. K. Aggarwal, "Object Recognition in Dense Range Images Using a CAD System as a Model Base", in Proc. of IEEE Conference on Robotics and Automation, Cincinnati, OH, May, 1990, pp 1858-1863.
2. F. Arman, and J. K. Aggarwal, "CAD-Based Vision: Object Recognition in Cluttered Range Images Using Recognition Strategies", Computer Vision Graphics Image Process: Image Understanding, Vol. 58, No. 1, pp 33-48, July 1993.
3. A. Johnson, and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes,"IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 21, No. 5, pp. 433 - 449, May 1999.
4. Y. Kim, and C. Haas, "A Model for Automation of Infrastructure Maintenance Using Representational Forms," J.of Automation in Construction, Elsevier, pp.57-68, October 2000.
5. J. McLaughlin, C. Haas, K. Liapi, S.V. Sreenivasan, and S. Kwon, "Rapid Human-Assisted Creation of Bounding Models for Obstacle Avoidance in Construction," in Proc. of 19th Annual ISARC, Gaithersburg, MD, Sept.23-25, 2002.
6. X. Lebegue, J.K. Aggarwal, "Automatic creation of architectural CAD models," in Proc. of the CAD-Based Vision Workshop, February 1994, pp 82-89.
7. T. Tsukiyama, "Understanding man-made environments using nonstructured lighting-3D world modeling for indoor mobile robots," Intelligent Robots and Systems, IROS '97, in Proc. of the 1997 IEEE/RSJ International Conference, Volume: 3, September 1997, pp 1250 -12577.
8. D. Simon, M. Hebert, and T. Kanade, "Real-time 3-D pose estimation using a high-speed range sensor," Robotics Institute, Carnegie Mellon University, CMU-RI-TR-93-24, November 1993.
9. G. Cheok, and W. Stone, "Non-intrusive Scanning Technology for Construction Assessment", in Proc. of the 16th International Symposium on Automation and Robotics in Construction (ISARC), Madrid, Spain, September 1999, pp. 645-650.
10. A. Johnson, R. Hoffman, J. Osborn, and M. Hebert, "A System for Semi-automatic Modeling of Complex Environments," in Proc. of International Conference on Recent Advances in 3-D Digital Imaging and Modeling, May 1997, pp. 213-220.
11. T. Feddema, and C. Little, "Rapid World Modeling: Fitting Range Data to Geometric Primitives," in Proc. of the IEEE International Conference on Robotics and Automation, Vol. 4, 1997, pp. 2807 - 2812.
12. B. Sabata, and J. K. Aggarwal, "Surface Correspondence and Motion Computation from a Pair of Range Images," Computer Vision and Image Understanding, Vol. 63, No. 2, pp 232-250, March 1996.
13. B. Sabata, F. Arman, and J. K. Aggarwal, "Segmentation of 3D Range Images Using Pyramidal Data Structures," Computer Vision Graphics Image Process: Image Understanding, Vol. 57, No. 3, pp 373-387, July 1993.
14. R. Hoffman, and A. K. Jain, "Segmentation and Classification of Range Images," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 9, no. 5, pp 608-620, September

1987.

15. A. Stentz, J. Bares, S. Singh, and P. Rowe, "A Robotic Excavator for Autonomous Truck Loading", in Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, pp. 1885 - 1893, 1998.
16. Cyra Technologies, Inc. 8000 Capwell Drive, Oakland, CA 94621, Tel: (510) 633-5009, www.cyra.com.
17. G. Cheok, R. Lipman, C. Witzgall, J. Bernal, and W. C. Stone, "Field Demonstration of Laser Scanning for Excavation Measurement," J. of Automation in Construction, Vol. 9, pp 463-477, 2000.
18. W. Stone, G. S. Cheok, R. Lipman, "Automated Earthmoving Status Determination," in Proc. of Robotics 2000. ASCE Conference on Robotics for Challenging Environments, Albuquerque, NM, February 28-March 2, 2000.
19. Y. Cho, C. Haas, K. Saidi, K. Liapi, and S. V. Sreenivasan, "Rapid Local Area Modeling for Construction Automation," in Proc. of 18th International Symposium on Automation and Robotics in Construction, Krakow, September 2001.
20. Y. Cho, C. Haas, K. Liapi, and S. V. Sreenivasan, "A Framework for Rapid Local Area Modeling for Construction Automation." J. of Automation in Construction, 11(6), pp. 629-641, 2002.
21. Occupational Safety and Health Standards - Excavation Final Rule, Occupational Safety and Health Admin., Fed. Register, U.S. Dept. of Labor, vol. 54, p 209, Washington, D.C., 1990.
21. C. Haas, M. Skibniewski, and E. Bundy, "History of Robotics in Civil Engineering," J. of Microcomputers in Civil Engineering, Vol. 10(5), pp371-381, 1995.
22. A. Johnson, O. Carmichael, D. Huber, and M. Hebert, "Toward a General 3D Matching Engine: Multiple Models, Complex Scenes, and Efficient Data Filtering," in Proc. of the Image Understanding Workshop, Monterey, 1998, pp 1097-1107.
23. C. Eberst, et al., "Robust Video-Based Object Recognition Integrating Highly Redundant Cues for Indexing and Verification" in Proc. of IEEE Int'l Conf. on Robotics and Automation, -3764, April 2000, pp 3757.
24. DistoMemo, Leica Geosystems Inc., CH-9435 Heerbrugg, Switzerland, Tel: 41 71 727 3131, www.leica -geosystems. com.
25. R. Duda, P. Hart, D. Stork, "Pattern Classification," Wiley-Interscience; 2nd edition, October, 2000.
26. X. Lebegue, J. K. Aggarwal, "Extraction and interpretation of semantically significant line segments for a mobile robot," in Proc. of Robotics and Automation, May 1992, pp 1778 -1785.
27. B.C. Vermuri, J. K. Aggarwal, "Representaion and Recognition of Objects from Dense Range Maps," IEEE Trans. on Circuits and Systems, Vol. CAS-34, No. 11, November 1987.
28. B.C. Vermuri, A. Mitiche, and J. K. Aggarwal, "Curvature-based Representation of Objects from Range Data," Image and Vision Computing, Vol. 4, no. 2, pp. 107-114, 1986
29. D. G. Schweikert, "An Interpolation Curve Using Spline under Tension," J. of Math. Phys., Vol 45, pp312-317, 1966

TABLE 1. EXPERIMENTAL RESULTS OF CUBOIDS

# of Pts	Angle View	Angle (edge X)	Angle (edgeY)	Angle (edge Z)	Time	Center Difference	diff. of width	diff. of length	diff. of depth
24	0	1.19	1.16	1.06	1.48	1.32	1.25	0.14	0.62
24	30	0.77	0.5	1.54	1.68	0.94	0.08	1.01	0.41
24	60	1.71	1.22	0.71	1.28	0.75	0.65	1.14	1.52
30	0	0.61	0.35	0.68	1.49	0.65	0.43	1.66	0.47
30	30	1.18	0.48	1.62	1.44	0.69	0.91	0.37	0.57
30	60	2.16	1.11	1.61	1.783	0.43	1.01	0.5	0.53
45	0	1.34	2.02	0.74	1.53	0.47	0.75	0.24	0.06
45	30	1.22	1.15	1.26	1.17	0.55	0.82	0.73	0.81
45	60	1.3	1.17	1.19	1.38	0.38	0.16	0.43	0.1

TABLE 2. EXPERIMENTAL RESULTS OF CYLINDER ALGORITHM

# of Points	Mean of STDV. of Radius				Mean of STDV. of Length				Mean of STDV. of Axis			
	L/D = 1.0	L/D = 1.5	L/D = 2.0	L/D = 2.5	L/D = 1.0	L/D = 1.5	L/D = 2.0	L/D = 2.5	L/D = 1.0	L/D = 1.5	L/D = 2.0	L/D = 2.5
10 pts.	0.46	0.19	0.18	0.12	1.10	1.30	1.07	0.72	12.71	2.39	2.32	1.75
20 pts.	0.58	0.15	0.16	0.12	1.44	0.80	1.06	0.71	14.11	2.30	2.31	1.62
30 pts.	0.62	0.14	0.14	0.11	1.22	0.77	0.68	0.71	18.80	2.26	2.27	1.60
40 pts.	0.86	0.14	0.14	0.08	1.68	0.73	0.77	0.56	21.99	2.17	2.27	1.54