

Human-Assisted Object Fitting to Sparse Range Point Clouds for Rapid Workspace Modeling in Construction Automation

by

Soon-Wook Kwon¹, Katherine A. Liapi², Carl T. Haas³, S.V. Sreenivasan⁴, and Jared T. McLaughlin⁵

ABSTRACT: In large-scale construction sites there are constant needs for rapid recognition and accurate measurement of objects so that on-site decisions can be made quickly and safely. Current methods involve full area laser range scanning systems that can produce very detailed models of a scanned scene, however the computational and data acquisition time that is required precludes the methods from being used for real time decision making. This paper presents algorithms to fit objects to sparse point clouds of measured data in a construction scene, that significantly decrease data acquisition time, and computational and modeling time. Two basic fitting and matching algorithms that address construction site material of cuboid and cylindrical shapes are discussed. Experimental results that indicate that the proposed algorithms assist an operator to create models of construction objects rapidly and with sufficient accuracy are also presented.

KEYWORD: CONSTRUCTION AUTOMATION; LASER RANGE FINDER; LEAST SQUARES METHOD; OBJECT FITTING; OBJECT MATCHING; WORKSPACE MODELING

1. INTRODUCTION

Using automated or semi-automated equipment on a large construction site requires rapid recognition and accurate measurement of objects in the workspace so that timely on-site decisions can be made. Most methods for modeling work environments rely on analyzing dense point cloud data, which requires computationally intensive processing, and usually takes much longer than the ongoing construction operation. Low accuracy in extracting objects from dense clouds is an additional limitation of full range scanning methods. Since most objects in a construction site are known and man made, they can be graphically generated and stored in object database as parametrically defined object classes [1]. 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 sensed data from 3D position sensors deployed in the construction environment [2][3].

The ability to extract models of real world objects in a construction workspace for

¹ Graduate Research Assistant, Dept. of Civil Eng., The University of Texas, Austin, TX 78712, swkwon@mail.utexas.edu

² Assist. Professor, Dept. of Civil Eng., The University of Texas, Austin, TX 78712, kliapi@mail.utexas.edu

³ Professor, Dept. of Civil Eng., The University of Texas, Austin, TX 78712, haas@mail.utexas.edu

⁴ Assoc. Professor, Dept. of Mech. Eng., The University of Texas, Austin, TX 78712, sv.sreeni@mail.utexas.edu

⁵ Engineer, Los Alamos National Lab, NM, jtmclau@sandia.gov

equipment operations from only a limited number of scanned points is a significant advantage of this approach over full range scanning methods that require intensive computational load because of range data processing for dense point clouds which consists of tens of thousand data points.

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 in a construction scene. 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 [4]. Cuboids can be used for fitting and matching structural objects such as columns, box-beams and walls and 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 each primitive:

1. Cuboid algorithm
2. Cylindrical object algorithm

2. EXPERIMENTAL SETUP AND HUMAN ASSISTED OBJECT FITTING AND MATCHING PROCESS

A single-axis laser range finder, a pan and tilt unit, and a personal computer were used for the experimental set up (Figure 1). The single-axis laser range finder (DistoMemo) that is mounted on the pan and tilt unit is designed not only for hand-held operation, but also for computer use through 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 accuracy of ± 3 mm. The step size of the tele-operated pan and tilt unit, 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 m in motion.

The sparse points cloud is acquired by

operator picking points to each object using single-axis laser range finder. The modeling process involves the following functions:

1. Select 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 fitting algorithm

Figure 2 displays a process diagram of these functions.

3. 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 during equipment controls [2][5].

For the rapid modeling of construction site objects from sparse point clouds two basic algorithms were developed that address construction site objects of cuboid and cylindrical shape. Since these two 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.

The following sections explain the fitting and matching methods which were developed and used for rapid workspace modeling:

3.1 Cuboid Algorithm

This section describes how to fit a

sparse points cloud to a cuboid's surfaces using the k-nearest neighbors and the least squares methods. There is an assumption of this process that three surfaces of cuboid should be visible in order to acquire data points.

3.1.1 Point Segmentation Using K-Nearest Neighbors Method

To find the nearest points for all measured points on a cuboid, a k-nearest neighbors algorithm was used. The algorithm finds the nearest two points by computing all the distances from a scanned point to all other points [6]. After determining two nearest neighbors for each scanned point, a group of three-point sets was found. Then, a normal vector for each three-point set was computed. By analyzing normal vectors, the scanned points were segmented by each cuboid surface.

3.1.2 Plane Optimization Using the Least Squares Fitting Method

The least squares method [7] was used for the best-planar fit of point sets on each surface of the cuboid after segmentation was applied.

Since in a planar regression, Y is to be regressed on two independent variables X and Z , a relationship, where both X and Z , are calculated as deviations from their means, was used:

$$E(Y_i) = \alpha + \beta_i \cdot X_i + \gamma \cdot Z_i \quad (1)$$

For any given combination of X_i and Z_i the expected yield $E(Y_i)$ is a point directly above the plane, shown as a hollow dot in Figure 3. The actual value of the component Y_i of an observed point is somewhat greater than its expected value and is shown as a solid dot lying on the plane. The difference between the observed and expected values of Y_i is shown by the error term e_i and thus the observed value Y_i is expressed as its expected value plus the error term e_i :

$$Y_i = \alpha + \beta_i \cdot X_i + \gamma \cdot Z_i - e_i \quad (2)$$

While moving along the x-direction, β_i is interpreted as the slope of the plane. In the same way γ is the subsidiary effect of z . To minimize the error sum of the squares a coefficient is used:

$$\sum_{i=1}^N (Y_i - \hat{Y})^2 = \sum_{i=1}^N (Y_i - \hat{\alpha} + \hat{\beta} \cdot X_i + \hat{\gamma} \cdot Z_i)^2 \quad (3)$$

Taking the partial derivatives of the above expression with respect to $\hat{\alpha}$, $\hat{\beta}$ and $\hat{\gamma}$, and setting them to zero, finally alpha, beta, gamma are found. Using this expression, the three optimized surfaces of the cuboid are computed (Figure 3).

After segmenting all scanned points by the three surfaces of the cuboid, the points were projected onto the optimized surface to compute dimensions.

3.1.3 Determining Intersecting Edges and Computing Dimensions

The three surface planes of the cuboid, from which range data were received, intersect at a point, and each two planes intersect at a line. The intersection of the two planes of the cuboid was found by solving the two linear equations representing the planes. After applying this for all three surfaces of the cuboid, the three edges of the cuboid were determined and matched. A vertex of the cuboid was also determined. Figures 4 and 5 show the results of point segmentation, and matching vertex process.

Once the three edges of the cuboid were defined, the dimensions of the cuboid were determined as follows: By computing the distances of all measured points on each surface to each one of the already defined edges of the same surface, the furthest point from each edge was found. The distances of the furthest points on the surfaces from the three intersecting edges represent the dimensions of the cuboid.

Figure 6 shows a fitted and matched model of an object after the application of the cuboid algorithm.

3.2 Cylindrical Object Algorithm

Four parameters are required for fitting and matching a solid cylinder: a scalar radius r ; an axis vector, a ; a center point to determine the axis vector, $c = (X_c, Y_c, Z_c)$ and a set of scanned points $g = \{(X_i, Y_i, Z_i)\}$ to find out the boundary of the cylinder. To determine the normal vector, the "k-nearest neighbors method" was used.

Then, by analyzing normals, the scanned points were segmented by surface (planar or curved). Subsequently, by projecting all points on the curved surface onto the planar surface, parameters r and c were estimated. The least squares method was also used to optimize the curved surface. The radius of the cylinder was found as the distance from the center of the circle to any point on the optimized curve. Projected points on the planar surface are considered end points of different chords in the circle and used to estimate its center \hat{c} . An initial estimate of the radius, \hat{r} , is found by $\hat{r} = \text{mean}(|\hat{c} - k'|)$ ($k' = \{\text{the points on the optimized curve of planar surface}\}$). Then the final values of a , c , and r are found by applying the least squares method to all scanned data (Figure 7).

4. EXPERIMENTAL RESULTS AND CONCLUSIONS

The fitting and matching algorithms discussed in this paper, are an integral part of a method that involves several other functions such as: human object recognition, collecting of range information, grouping of scanned points, and computing dimensions to final fitting and matching. A basic feature of the method is that it takes advantage of 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.

Experiments were conducted to determine the efficiency of the human assisted modeling method. The algorithms, which are based on the least squares method, were found to be useful for modeling construction objects of cylindrical and cuboid shapes. They were applied to determine the width, depth, and height of cuboids, and the diameter, and height of solid cylinders including the location and orientation.

Table 1 shows an example of experimental results of a cuboid fitting and matching process. The test results of the algorithms present approximately less than 1-degree angular deviation between model and real objects' axis.

Respectively in all tests the size difference between the modeled and the actual object's surfaces is 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 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.

5. REFERENCES

- [1] Cho, Y., Haas, C., Saidi, K., Liapi, K., and Sreenivasan, S.V.(2001), "Rapid Local Area Modeling for Construction Automation" Proc., 18th International Symposium on Automation and Robotics in Construction, Krakow, September
- [2] Kim, Y.S. and Haas, C. (2000) "A Model for Automation of Infrastructure Maintenance using Representational Forms", Journal of Automation in Construction, Vol.10, No.1.
- [3] Cho, Y. (2000), "Human-Assisted Rapid Workspace Modeling for Construction Equipment Operations," PhD Dissertation, The University of Texas at Austin
- [4] Johnson A., Hoffman R., Osborn J., and Hebert M.(1997), "A System for semi-automatic Modeling of Complex Environment" Proc. IEEE International Conference on 3-D Digital Imaging and Modeling, Ottawa, May
- [5] McLaughlin, J. T. (2002), "Rapid Human-Assisted Creation of Bounding Models for Obstacle Avoidance in Construction," Master's Thesis, The University of Texas at Austin
- [6] Duda R. O., Hart P.E., and Stork D.G. (2001), "Pattern Classification" Wiley Inter-Science
- [7] Duran B.S., and Amir-Moez A.R. (1973) "Linear Algebra of The Plane", Texas Tech University

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.



Figure 1. Experimental Setup

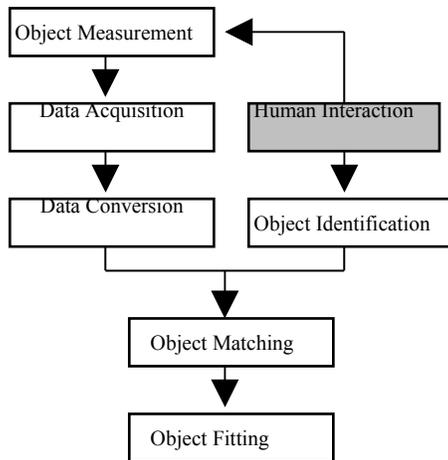


Figure 2. Fitting and Matching Process

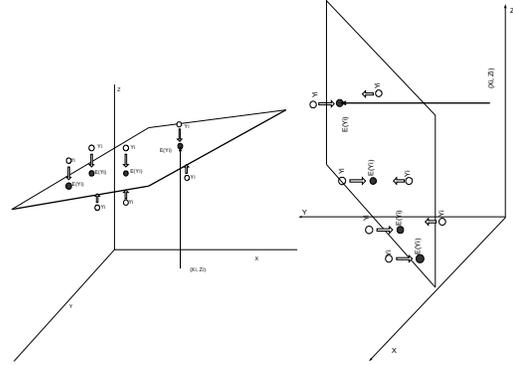


Figure 3. Surface Optimization

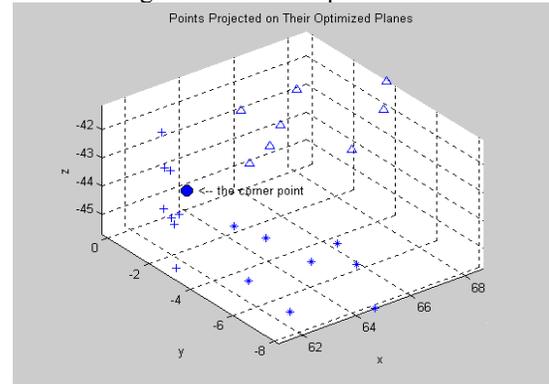


Figure 4. Matching Points and Segmentation

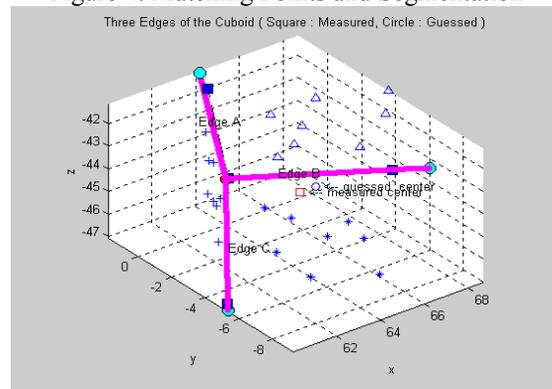


Figure 5. Three Edges of a Cuboid and its Centroid

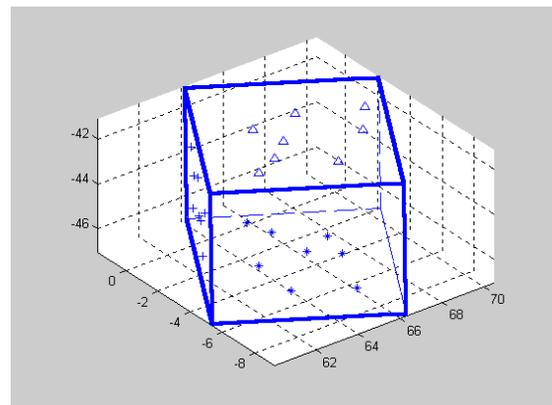


Figure 6. Fitted and Matched Cuboid

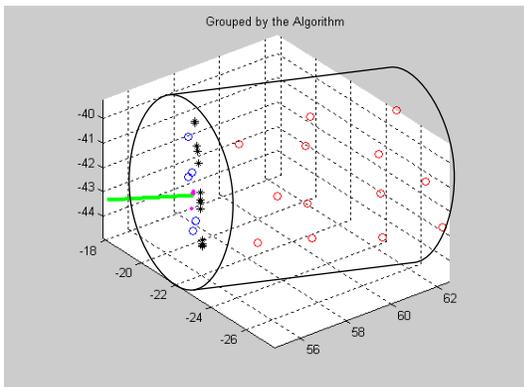


Figure 7. Fitted and Matched Cylinder

Table 1. Test Results of the Cuboid Algorithm

Matching Point (vertex)		Modeled Object	Actual Object	Deviation
Object 1	x	57.749	57.816	0.067
	y	11.697	11.733	0.037
	z	-35.447	-35.464	-0.017
Object 2	x	59.981	60.037	0.056
	y	-5.203	-5.136	0.067
	z	-41.370	-41.543	-0.173
Object 3	x	59.967	60.028	0.061
	y	-5.182	-5.108	0.073
	z	-41.354	-41.492	-0.138
Object 4	x	59.918	60.032	0.114
	y	-5.170	-5.109	0.061
	z	-41.210	-41.621	-0.411
Angular deviation between edges	Edge A			1.089
	Edge B			1.824
	Edge C			0.927
Measuring + Computing time	pts	Measuring (sec.)	Computing (sec.)	
	16	30.00	5.87	35.87
Object 1	16	30.00	5.87	35.87
Object 2	18	50.00	5.66	55.66
Object 3	22	60.00	6.48	66.48
Object 4	26	80.00	5.60	85.60