Construction Object Identification from LADAR Scans: An Experimental Study Using I-Beams

David E. Gilsinn[1]
Geraldine S. Cheok[2]
Christoph Witzgall[1]
Alan Lytle[2]

[1]Mathematical and Computational Sciences Division
Information Technology Laboratory

[2]Materials and Construction Research Division
Building and Fire Research Laboratory

NISTIR 7286

U. S. Department of Commerce
Technology Administration
National Institute of Standards and Technology
Gaithersburg, MD 20899

DISCLAIMER

Certain trade names and company products are mentioned in the text or identified in an illustration in order to adequately specify the experimental procedure and equipment used. In no case does such an identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are necessarily the best available for the purpose.

ABSTRACT

Laser Scanning devices (a.k.a. LADAR for Laser Detection and Ranging) are primarily used in construction projects to capture as-built data for documentation and re-engineering. These systems can rapidly generate unstructured point clouds of scene data within the field of view and line-of-sight of the LADAR. The Construction Metrology and Automation Group (CMAG) at NIST is interested in using these high-resolution scanned data to identify and locate construction objects of interest. The test operation is large-scale pick-and-place of structural steel, where the data from the scans are used to identify the general pose, position and orientation of a target object for initial positioning of a robotic crane. Close-in control of the crane using high frame-rate flash LADAR is a separate but related area of CMAG research.

This initial study describes an experiment in which an I-beam on a concrete floor surface is scanned, and the resulting point cloud data are used to calculate its pose. Multiple scans of two sizes of I-beams were taken with the I-beam placed at various angles relative to the scanner. Two approaches for segmenting potential target objects are described: binning and triangular irregular networks (TINs). Using the method of principal components analysis, the pose of potential target objects is determined. Bounding boxes are then formed around these objects and compared to an ideal bounding box generated from the known geometric specifications of the I-beam of interest. A measure of best fit is used to determine which scanned object most closely fits the bounding box of the ideal I-beam. A separate laser-based site measurement system (SMS) was used to measure points on the I-beams to form reference data for estimating the closeness of fit of computed pose of the I-beam to measured pose of the I-beam. Three spheres were located in the scan field as a means of registering the scan and SMS axes to one another.

Keywords: binning, bounding boxes, LADAR, object identification, pose, principal axes, triangulated network, principle components analysis, irregular networks.

CONTENTS

1. INTRODUCTION

Concerns regarding inefficiencies in component, material, and trades tracking on construction sites have been repeatedly voiced at National Institute of Standards and Technology (NIST) workshops (Automated Steel Construction [64], Data Exchange Standards at the Construction Site. Improved asset tracking systems will enable both leaner construction and enhanced security, as well as lay the foundation for levels of automation envisioned in an Intelligent and Automated Construction Job Site [24]. The combination of LADAR technology, real-time object recognition, automatic identification, and other tracking technologies (e.g., GPS, Ultra-Wide Band, etc.) provide powerful potential mechanisms for assessing real-time status of construction site operations and future autonomous construction systems.

The Construction Metrology and Automation Group (CMAG) of NIST has an ongoing research effort in Construction Object Recognition and Tracking. One element of this project is focused on determining the pose of an object given a point cloud (in the form of x, y, z data) of a scene as obtained from a high resolution LADAR. One objective of these efforts is to determine the pose of a known, targeted object, without user intervention, to provide initial guidance information to an automated crane [74]. Close-in control of the crane for object pick-up and docking using a high frame-rate flash LADAR is a separate but related area of CMAG research.

Automated object recognition from LADAR point cloud data is an ongoing research area, see for example Matoba, et al. [67], Zheng, et al. [91], Weiss and Ray [80], Whitaker and Gregor [82] and Greenberg et al. [30] and other references in those papers. The current experimental effort has concentrated on developing an algorithm to segment the data (i.e. remove data not belonging to potential objects of interest) and to determine the pose of a targeted object, an I-beam in this case. The general project strategy is to obtain experimental LADAR scans of a known object, apply two different algorithms to the data to predict the I-beam pose, and finally use an independent reference system to measure the actual I-beam pose to assess the performance of the pose determination methods.

There were four aspects to this experiment. First, multiple LADAR scans were taken of a simple scene which included an I-beam. Second, two approaches were then taken to segment potential objects of interest in the scan field. The first method employed binning the scan data into voxels while the second method relied on creating a triangular irregular network (TIN). Both methods subsampled the scan data and provided a mechanism to isolate potential objects. The third aspect included creating bounding boxes around potential objects by applying principal components analysis to the segmented scan data. These bounding boxes were compared to an ideal bounding box derived from the I-beam specifications to identify the target object. Finally, the calculated object pose was compared to the measured object pose. The measured object pose was obtained using a laser-based site measurement system (SMS) that was able to locate the I-beam more accurately than with the LADAR. The transformation matrix required to tranform the coordinates in the SMS frame to the LADAR frame was calculated using three spherical targets and the location of the scanner.

This report is organized as follows. Section 2 provides a brief introduction to LADARs and the specifications of the LADAR used for the experiment. Section 3 describes the LADAR settings and scene/scanner geometry. Section 4 describes two methods of segmenting objects, the use of principal components analysis to determine object pose, and the use of bounding boxes to identify the segmented objects. Section 5 discusses three methods of determining the center and radius of reference spheres. Section 6 presents comparative results of the object identification and pose determination using the two segmentation techniques. Section 7 presents some conclusions. The appendices include various tables of data used in the analysis.

2. LADAR SYSTEM

Laser radars have been in use for many years. For example the theory of 2-D pulsed imagers is described in Enders and Shapiro [22]. These early devices returned a range and signal intensity. LADARs were then introduced that provided angular orientation, range, intensity of returned signal, and more recently, color. Early problems and sources of error for these laser radars were studied in the early 90s. See for example Hebert and Krotkov [34] and Hashemi, et al. [33]. A discussion of the design and operation of 3D scanners used in space applications was given in Beraldin et al. [4, 5]. The design and operation of a laser scanning system was also given by Golnabi [27]. An assessment of the performance of a 3D whole body imaging system was given by Marshall et al. [66]. A critical review of laser ranging was reported by Amann et al. [1]. For a review of LADAR technology see Stone et al. [76]. In this section we give a brief discussion of LADAR operation and present the characteristics of the LADAR used in the current study.

2.1 GENERAL INFORMATION

LADAR configurations vary but in general all scanning systems involve the use of an emitted optical signal, a system to point the emitted signal throughout the system's field-of-view, and an optical receiver element to detect and process the returned signal. The optical signal is normally generated from a laser, a laser diode, or an LED (Light Emitting Diode), and the typical pointing system is either a mechanical scanning (e.g., rotating polygonal mirror) or beam-steering system (e.g., acousto-optic, electro-optic). The receiving element is most often integrated in the optical train with the emitter. The received signal can provide spatial, reflectance, and color information. Spatial data is typically provided as r, θ, φ from the instrument origin and post-processed as x, y, z data coordinates in the instrument frame. Range information is obtained by either 'clocking' the signal return in pulsed time-of-flight (TOF) systems or by comparing phase differentials for a modulated emitted signal and its return. The pointing vector data - θ and φ – are derived from scanning mechanism data such as encoder values. Reflectance (intensity) data can be active-illumination reflectance, passive (white-light) reflectance, or both. Color data can be collected from either a co-boresighted CCD (Charge Coupled Device) or a separate imaging system co-registered with the LADAR.

2.2 CHARACTERISTICS OF THE LADAR USED IN THE EXPERIMENT

The manufacturer's specifications for the LADAR used in this study are provided in Table 2.1.

Table 2.1.Manufacturer's Specifications for LADAR Instrument Characteristics

Instrument Characteristic	Manufacturer's Specified Value
Measurement Range[a]	2 m to 350 m for natural targets, $\rho \geq 50$ % 2 m to 150 m for natural targets, $\rho \geq 10$ %
Measurement Accuracy[b] (1 σ standard deviation)	±2.5 cm, in the worst case ± 10 cm[d]
Vertical Field-of-View	80º
Horizontal Field-of-View	0º to 333 º

[a] Typical values for average conditions. In bright sunlight, the operational range is considerably shorter than under an overcast sky. At dawn or night the range is even higher.
[b] Standard deviation, plus distance depending error ≤ 20 ppm.
[c] The conditions which constitute worst case is not specified.

2.2 PROBLEMATIC ASPECTS OF LADAR DATA

2.2.1 Beam Divergence Problem

The divergence or spread of a beam can produce distortions. An earlier study by Gilsinn et al. [5] indicated that for at least one LADAR, the dispersion could range from 50 mm at 20 m from the LADAR, to 100 mm at 40 m, all to way to about 250 mm at 100 m. This dispersion led to defocusing of the scanned images. A method using image deconvolution to correct for the resulting distortion was investigated. The problem of beam divergence was also noted by Hebert and Krotkov [34] where they point out that the beam subtends a solid angle. They call the area of the beam projected on the target surface the *footprint* of the beam and indicate that every point in the intersection of the footprint and target contribute range values and intensities to the final range and intensity value of the measurement. The effects of beam divergence are alleviated for instruments with smaller beam sizes and those which incorporate methods to reduce beam divergence. The report by Stone et al. [76] includes a discussion of the affects of beam divergence. They point out that due to the divergence of the beam the photons associated with a single "pixel" in a LADAR frame may represent significantly differing range values. This effect can be used for positive good, for example, if multiple photon hits of a single "pixel" beam were recorded then it is possible to penetrate foliage. The divergence of the beam however leads to the problem of mixed pixels for some LADARs.

2.2.2 Mixed Pixels

Errors can occur at locations when the beam footprint is split and there are multiple returns for a given scan direction. For example, this can occur when scanning an object with a sharp edge. Imagine a scenario where the scanned object is a box positioned on a floor. At the box edge, part of the beam hits the box and returns and part of the beam continues on to hit the floor. Both returns are sampled. Various strategies exist for calculating range from the returned energy. These include first pulse, last pulse, or integrating all pulses (with appropriate filtering for determining a valid pulse). Another approach would be to provide the user with all pulse data (i.e. multiple returns per pixel). The LADAR used in this experiment calculates the range by integrating over the entire projected footprint. This leads to producing a 3D data point that is actually located somewhere between the box and the floor. These points are called *mixed pixels* or sometimes *phantom pixels*. Figure 2.1 shows the point cloud for a box and the mixed pixels formed off of its top and side edges.

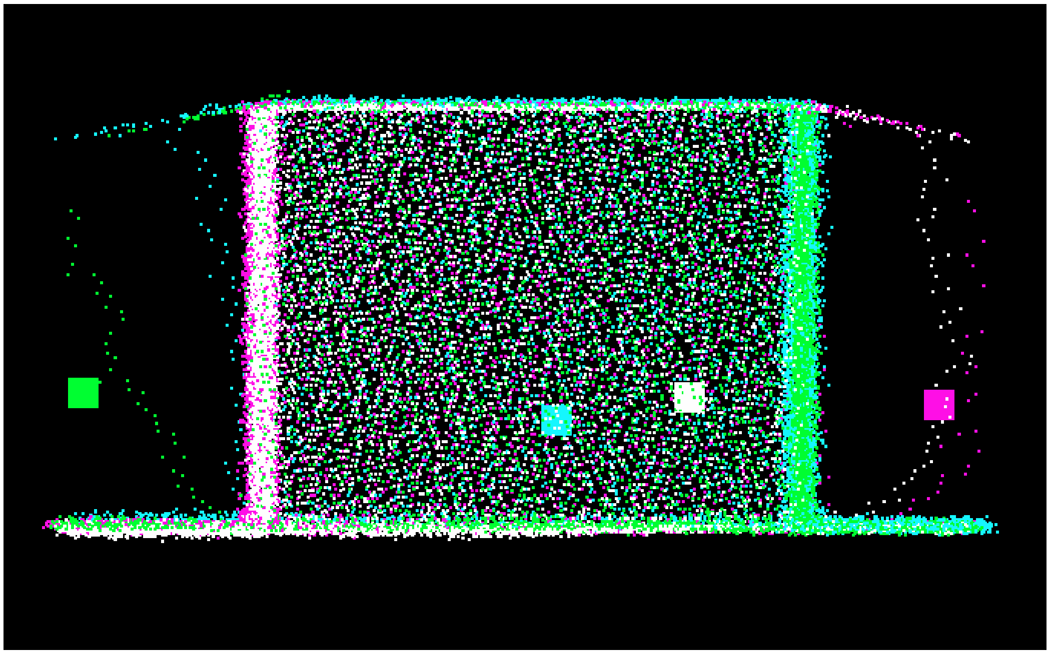


Figure 2.1. This image shows the line of mixed pixels extending from the top and side edges of the box.

The mixed pixels complicate the image recognition problem because sharp edges are difficult to detect. Therefore, such standard image processing tools as edge detection or edge following methods cannot be reliably applied without prefiltering the data to remove mixed pixels. For a discussion of this problem and other related laser radar problems see Hebert and Krotkov [34] and Stone et al. [76].

3. LADAR SCAN CONFIGURATIONS FOR THE STUDY

A series of scans were conducted to determine the uncertainty of the pose calculations, the influence of point density on pose determination, the influence of point-of-view of the scanner relative to the object on pose determination, and the ability of the algorithm to differentiate between two objects of similar shape. In these experiments, an I-beam in an uncluttered environment was scanned using the LADAR described in Section 2.2. In order to reduce the point cloud size the data set was visually trimmed to an area-of-interest that included the I-beam and the the tripods holding the reference spheres (see Figure 3.1b).

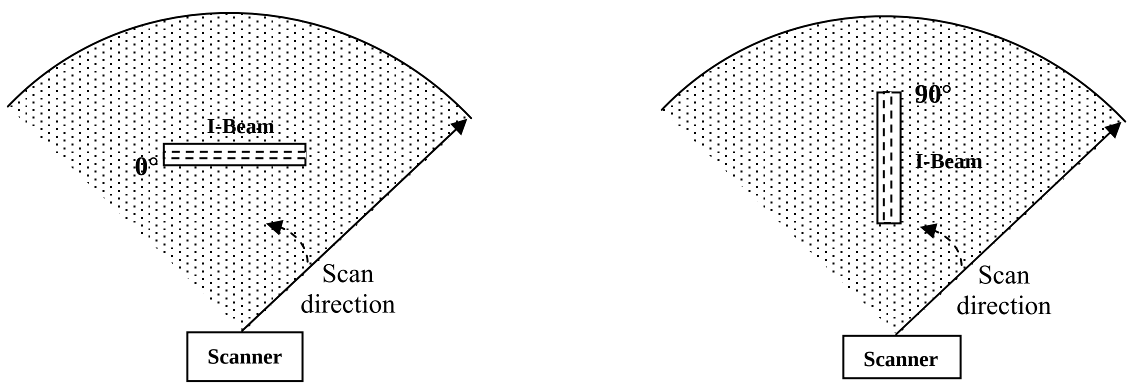
The dimensions of the two I-beams used in the experiments are given in Table 3.2. The I-beam was oriented at 0°, 30°, 45°, 60°, and 90° relative to the LADAR (see Figure 3.1a). At an orientation of 0°, the center of the I-beam was approximately 7 m from the scanner. This study used three settings of the angular increments. The highest density of points was obtained with an incremental angle of 0.072° (0.08 gon). 1 gon is an angular measure the equivalent of 0.9^{o}. The medium density of points was obtained at an incremental angle setting of 0.108° (0.12 gon) angular increments, and the lowest density was obtained at an angular increment setting of0.180° (0.20 gon). The further an object is from the scanner the fewer points will hit the target object. Table 3.1 gives estimates of the point increments at 5 m, 10 m, 20 m, and 40 m from the LADAR for the three point densities. Additionally, at each orientation and point density level, 3 scans were obtained to quantify the uncertainty associated with the pose calculation for the given orientation and point density. This resulted in a total of 90 scans being obtained (see Table 3.3). In many of the tables in this report we use the word *density* to refer to the density of sample points generated by the increments above. In particular high density refers to points sampled at 0.072° (0.08 gon), medium density at 0.108° (0.12 gon), and low density at 0.180° (0.20 gon).

Table 3.1. Theoretical scan point spacing as a function of range

Angular Increment ° (gon)	Distance from LADAR (m)			
	5	10	20	40
0.072 (0.08)	6 mm	13 mm	25 mm	50 mm
0.108 (0.12)	9 mm	19 mm	38 mm	75 mm
0.180 (0.20)	16 mm	31 mm	63 mm	126 mm

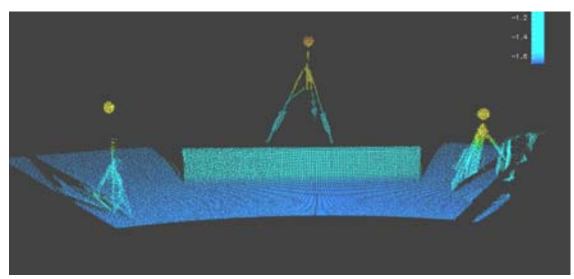
Table 3.2. I-Beam Dimensions.

	Length m (ft)	Depth, *d* mm (in)	Flange width, b_f mm (in)	Flange thickness, t_f mm (in)	Web thickness, t_w mm (in)
I-Beam A	3.050 (10.005)	419.1 (16-1/2)	179.4 (7-1/16)	18.3 (23/32)	11.1 (7/16)
I- Beam B	2.134 (7.000)	363.5 (14-5/16)	254.0 (10)	20.6 (13/16)	13.5 (17/32)



a. I-beam orientation.

b. I-Beam at 0°. Spheres used for registration.



c. Point cloud of a scan.

Figure 3.1. Experimental set-up. The data in Figure 3.1c was trimmed so that it fell between a radius of 3 m and 10 m around the scanner.

Table 3.3. Experimental Test Matrix.

I-Beam A					
Ang. Incr.	**I-Beam Orientation (°)**				
	0	**30**	**45**	**60**	**90**
0.072° (0.08 gon)	A0-8gon-1 A0-8gon-2 A0-8gon-3	A30-8gon-1 A30-8gon-2 A30-8gon-3	A45-8gon-1 A45-8gon-2 A45-8gon-3	A60-8gon-1 A60-8gon-2 A60-8gon-3	A90-8gon-1 A90-8gon-2 A90-8gon-3
0.108° (0.12 gon)	A0-12gon-1 A0-12gon-2 A0-12gon-3	A30-12gon-1 A30-12gon-2 A30-12gon-3	A45-12gon-1 A45-12gon-2 A45-12gon-3	A60-12gon-1 A60-12gon-2 A60-12gon-3	A90-12gon-1 A90-12gon-2 A90-12gon-3
0.180° (0.20 gon)	A0-20gon-1 A0-20gon-2 A0-20gon-3	A30-20gon-1 A30-20gon-2 A30-20gon-3	A45-20gon-1 A45-20gon-2 A45-20gon-3	A60-20gon-1 A60-20gon-2 A60-20gon-3	A90-20gon-1 A90-20gon-2 A90-20gon-3
I-Beam B					
Ang. Incr.	**I-Beam Orientation (°)**				
	0	**30**	**45**	**60**	**90**
0.072°	B0-8gon-1 B0-8gon-2 B0-8gon-3	B30-8gon-1 B30-8gon-2 B30-8gon-3	B45-8gon-1 B45-8gon-2 B45-8gon-3	B60-8gon-1 B60-8gon-2 B60-8gon-3	B90-8gon-1 B90-8gon-2 B90-8gon-3
0.108°	B0-12gon-1 B0-12gon-2 B0-12gon-3	B30-12gon-1 B30-12gon-2 B30-12gon-3	B45-12gon-1 B45-12gon-2 B45-12gon-3	B60-12gon-1 B60-12gon-2 B60-12gon-3	B90-12gon-1 B90-12gon-2 B90-12gon-3
0.180°	B0-20gon-1 B0-20gon-2 B0-20gon-3	B30-20gon-1 B30-20gon-2 B30-20gon-3	B45-20gon-1 B45-20gon-2 B45-20gon-3	B60-20gon-1 B60-20gon-2 B60-20gon-3	B90-20gon-1 B90-20gon-2 B90-20gon-3

4. OBJECT IDENTIFICATION

Pose determination in this study refers to the ability to recognize a known object in a scan scene and determine its location and orientation relative to the 3D scanner. In this work, target objects in a scanned scene are represented by unstructured (x, y, z) points in 3D space. These are sometimes referred to in the literature as *range images*. The difficulty of recognizing the target object depends on the size and complexity of the scanned scene, the number of objects in the model database, and the amount of *a priori* information about the scene. In the simplest scenes, objects are un-occluded and appear in a few standard positions. For a literature review of pose determination using range data see Greenspan [31]. Hoover et al. [38] did an experimental comparison of range image segmentation algorithms up to 1996. For a discussion on recognizing 3D objects using surface descriptions see Fan et al. [23], Liang and Todhunter [59], Huttenlocher and Ullman [43], Wurster et al. [88], Johnson, et al.[46], Caelli et al. [13], Zhang and Hebert [89], Johnson and Hebert [47]. Krishnapuram and Casasdent [52] introduced a technique for determining location and orientation of general three-dimensional objects from a single range image. Kim and Kak [50] introduced a graph matching technique for comparing range objects with model objects. Automatic target recognition systems have been studied by Verly et al. [79], Green and Shapiro [29], Zheng et al. [91], Hutchinson et al. [42], Greenberg et al. [30]. Probabilistic methods for object recognition have been used by Wheeler and Ikeuchi [83]. Surface-matching by harmonic maps have been used by Zhang and Hebert [89]. Edge detection has been used by Jiang and Bunke [44]. Cooper et al. [16] studied information theoretic measures, entropy and mutual information, as performance metrics for object recognition. Mirmehdi et al. [68] introduce feedback control strategies for object recognition. Matoba et al. [67] projected 3D objects onto an array of 2D images. Weiss and Ray [80] use geometric invariants to reduce complexity of object recognition problems. Even a quick review of the literature indicates that many of the current techniques depend on an identification of edges, points, and surfaces in the scanned objects and comparing these with similar features in database models. However, many of the previous references required dense object sampling not necessarily available to LADARs at long range. LADAR images often do not provide clear edge contours so that surface features can be identified. It is often necessary to rely on more elementary techniques to gauge an approximate object structure automatically from point clouds.

Once objects have been segmented within point clouds a second task involves modeling them. There is a vast literature on modeling 3D objects and we can only give a sampling of it here. This list covers many of the surface reconstruction methods developed over the last ten years or so. Of those that describe modeling of range data, most give examples of relatively small, nearby objects, such as statues or somewhat standard toy shapes. A few papers described modeling of 3D data acquired from longer range sampling of buildings and entire rooms. The papers descrbing modeling techniques are the following: Hoppe et al. [39], Welch and Witkin [81], Krishnamurthy and Levoy [51], Zhao and Mohr [90], Lindstrom et al. [60], Leonardis et al. [57], Luebke and Erikson [63], Cignoni et al. [15], Hirsch et al. [37], Dorai and Jain [20], Bors and Pitas [11], Witzgall et al. [85, 86, 87]. There are a number of papers that describe modeling techniques applied to range data. Many, however, involve scans of relatively small objects or modeling of available point cloud datasets. The following references describe some of these methods: Sourlier and Bucher [75], Bajaj, et al. [2], Curless and Levoy [19], Bernardini et al.

[8], Lu and Yun [61], Juarez et al. [48], Boughorbal et al. [12], Chatzis and Pitas [14], Levy and Lindenbaum [58], Marshall et al. [65], Barhak and Fischer [3], Whitaker and Gregor [82]. There are a few references that describe modeling of point clouds obtained from large scale scans, such as buildings and statuary. The following references give some of those details: Beraldin et al. [6], Huber et al. [41], Pollefeys et al. [72], Kanamaru et al. [49], El-Hakim [21].

In this study, the scene was constrained to contain only a few objects in addition to the target object. There were no objects of similar dimension to the target object and the floor was smooth. Data points outside the volume of interest (outside the furthest spherical registration target) were manually removed. A particular difficulty faced in this study was alluded to in the discussion of problems related to mixed pixels in Section 2.4. Edges of objects are not clearly delineated. This is seen in Figure 4.1, which shows an I-beam with three tripods around it. One of them is faintly seen in the llower middle of the figure.

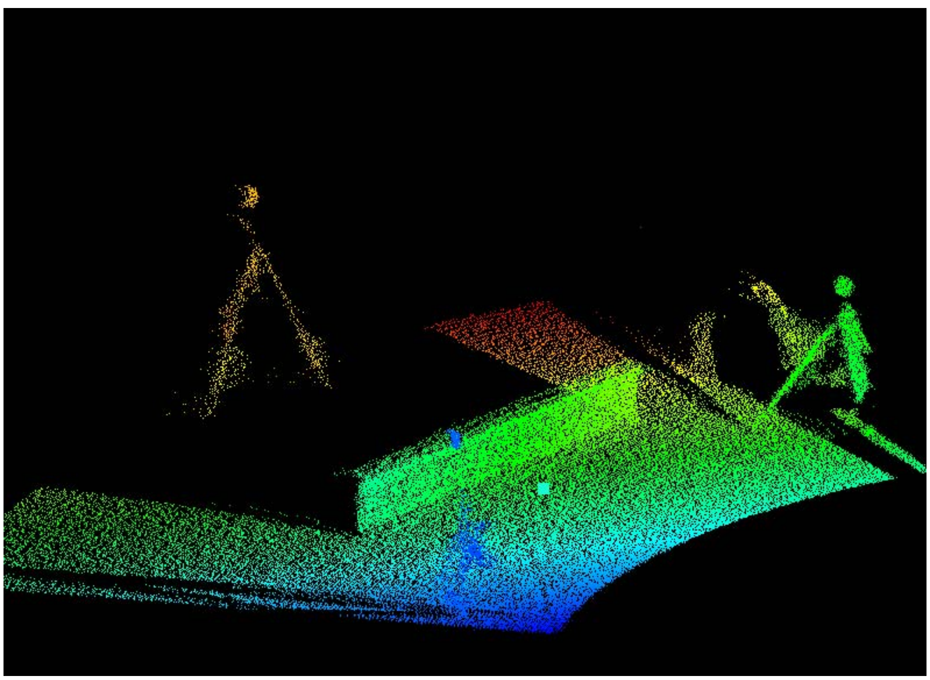


Figure 4.1. Sample 3D scan data set from the LADAR. Eyepoint rotated to give perspective view. Note that objects in the field of view are not well delineated.

For this study we do not use feature identification, but we rely on the use of bounding boxes around segmented data to identify an object. The object is segmented using two approaches. In the first, we use voxel (for definition see Section 4.1.1) binning and in the second we use triangulation. Once potential objects are segmented, we apply a more global idea of determining the principal axes of potential objects and enclosing them in bounding boxes. These box polygons are then matched against box polygons around data set objects using a goodness-of-fit metric to identify the object. These ideas are expanded in this chapter. An earlier description of some of the methods used in this object recognition study is given in Gilsinn et al. [4].

4.1. OBJECT SEGMENTATION

4.1.1 Binning

In order to make the scanned data sets more manageable, the data points are accumulated into volume cubes or *voxels* that encompass the entire range of the data set. This process is called *binning*. The number of voxels used to model a scanned volume can be determined in multiple ways. The selection tends to be a compromise between the compute time to analyze the resulting mesh and the size or resolution of each voxel. The selection of distances between voxel centers would depend on the nature of the application. In some cases centimeters might be acceptable whereas in others millimeters might be required. In either case, the number of voxels in the mesh must be controlled so that compute time does not become too great. For example, in the analysis of the data obtained in the current study a mesh of (60 x 60 x 60) voxels was found to be sufficient.

In the binning algorithm, the size of each voxel is calculated as a triplet (*dx*, *dy*, *dz)*, e.g.,

$$dx = \frac{(x\max - x\min)}{(\#\ partitions\ in\ x)}.$$

Each (*x*, *y*, *z*) point is associated with an (*i*, *j*, *k*) value for each bin. In terms of *x*, for example, this is computed for each point *r* of the scanned data as $i(r) = \text{fix}[(x(r) - xmin)/dx] + 1$, with due consideration of the boundary voxels. Similar index calculations are performed for *j*(*r*) and *k*(*r*) for each *N* points. Once the (*i*(*r*), *j*(*r*), *k*(*r*)) indices are computed for each point, the bin count (number of points) per voxel can be determined. The bin count of all voxels is initially set to zero. Due to the binning, many (*x*, *y*, *z*) points will have the same (*i*(*r*), *j*(*r*), *k*(*r*)) indices. The count algorithm iterates by starting with index triplet (*i(1)*, *j(1)*, *k(1)*) and finding all matches of this triplet by a simple linear search. The number of matches, *M*, gives the bin count for the (*i*(*1*), *j*(*1*), *k*(*1*)) voxel. These matched indices are then eliminated from the (*i*(*r*), *j*(*r*), *k*(*r*)) list and a new allocated list is formed of length *N-M*. The old list is deallocated. This reduces the number of (*i*(*r*), *j*(*r*), *k*(*r*)) triplets that need to be linearly searched. The algorithm then begins again with the new (*i*(*1*), *j*(*1*), *k*(*1*)) triplet, finds the matches, counts them for that voxel and then eliminates those triplets. The process continues until there are no more (*i*(*r*), *j*(*r*), *k*(*r*)) triplets. All voxels would then have a bin count associated with them. Many would have a bin count of zero (empty). For a further discussion of 3D image representation see Nikolaidis and Pitas [71].

The binning process was selected instead of surface fitting for several reasons. It allows objects to be modeled by relatively simple polygon structures which are appropriate for I-beam identification. Polygon models also provide sufficient structure for alignment and they allow fast comparisons with similar polygon models of the ideal structures by only matching a relatively few vertices.

Once the data has been binned into a workable set of voxels, objects can be isolated. First, however, an algorithm is used to eliminate as many mixed pixels as possible. The algorithm is based on the idea that those columns with floor hits or phantom pixels would have a number of voxels with small bin counts and in general very few voxels connected to them in the column. It should be noted that voxels can have more than one phantom pixel in them depending on voxel resolution. By a connected voxel, we mean one with a nonzero bin count either directly above or directly below. These are also referred to as *neighbors*. A vertical string of voxels is defined to be the set of non-zero voxels in one column that are direct neighbors of each other. The most significant vertical voxel string in a column is the longest string in each column. This is determined by examining each vertical column of voxels and associating a value of “1” in a buffer for that column if a bin is non-zero, otherwise the buffer is set to “0”. The string length and maximum string length for the column are initialized to “0”. Each voxel up the column is examined. If the buffer value is “1”, then the string length is incremented and the maximum string length is set to the current string length if it is longer than a previous string length in the column. Once the buffer becomes zero the current string length is reset to zero and the next voxel is examined until the column is finished. The maximum string length per column is automatically produced. Also, the maximum overall string length is the maximum of all of the column string lengths. Columns that include those voxels that are to be eliminated are determined by comparing the maximum string length up a column with the maximum overall string length. If the maximum column string length is less than a prescribed fraction of the maximum overall string length then the bin counts in that column are all set to “0”. Currently, that factor is selected by the user, but future algorithm enhancements will include a more general factor determination. Columns with floor hits or phantom pixels usually have maximum column string lengths of one or two as opposed to maximum column string lengths of 40 or 50 or more for tall objects, again depending on voxel resolution.

Once the outlier voxels, such as phantom pixels and ground hits have been eliminated, entities called *objects* can be identified. The basic premise of the object segmentation portion of the overall algorithm is that objects are made of neighboring voxels. Thus, this portion of the algorithm accumulates neighboring voxels into object structures. The sub-algorithm used here is based on one proposed in Nikolaidis and Pitas [71] for calculating the volume of connected components in 3D. The process starts by constructing a three dimensional array, called the *mask array*, of the same size as the voxel array and assigning the value “1” to the mask element (i, j, k) associated with the voxel indexed (i, j, k) if that voxel element is non-zero, otherwise the mask array element is set to “0”. The segmentation proceeds by first finding a non-zero mask element and setting it to “0” after storing its voxel information on a stack. It then proceeds with setting to “0” each of the mask array elements for every neighboring non-zero mask element that is found, and then placing the associated voxel information associated with each of those mask elements on a temporary holding stack. For those neighbors, their neighbors are examined in an outward expanding group of neighboring voxels. When there are no more non-zero neighboring mask elements to be set to “0”, the stack is unloaded into an object structure. Setting the mask array elements to “0” for all of the voxels in an object is necessary because when the algorithm goes back to the mask array to find another object, it won’t find a mask element associated with the object just created.

4.1.2 Triangulation (TIN) methods

The acronym "TIN" stands for "Triangulated Irregular Network", an approach to representing given 3D point clouds by triangulated or TIN surfaces which consists of a connected set of triangles in 3D. A TIN surface thus has vertices and edges, namely, the corners or vertices and the edges of the triangles. The general understanding of the TIN approach, however, is based on a more restricted class of triangulated surfaces. It operates with triangulated surfaces which are also graphs, respectively, of bivariate functions $z = z(x, y)$, and which are usually defined over a rectangular area or "map" in the x,y-plane. As a consequence, each point (x, y, z) on the TIN surface is the sole point with the projection or "footprint" (x, y) in the x, y-plane. Surfaces with the latter property are referred to by various authors as "parametric" or "2.5D". Note that such surfaces may not have switchbacks or overhangs.

It follows, also, that the projections of the 3D triangles of the TIN surface into the map are 2D triangles, which cover the map without overlapping and thus define a "triangulation" of the map area. The 2D vertices of the triangulation are the footprints of the 3D vertices of the TIN surface, and the triangulation thus defines which of these vertices belong together as corners of 3D triangles of the surface. Similarly, the triangulation defines which 3D vertices are neighbors to each other. The triangulation of the 2D vertices can thus be viewed as an irregular net over which a triangulated surface is defined.

In typical applications, and also in the work reported here, the vertices of the TIN surface are – aside from the map corners – actual data points (x_i, y_i, z_i) selected from a given point cloud. The construction of a TIN surface then amounts to constructing a triangulation of their footprints (x_i, y_i). For the purposes of this work, the "Delaunay triangulation" has been chosen. A triangulation of S, where S is a set of 2D points, is called Delaunay if the following property holds. Any three points in S describe a triangle if and only if the interior of their circum-circle does not contain points in S. The TIN approach is widely understood as Delaunay based.

Many algorithms for constructing Delaunay triangulations are available. At NIST, an adaptive version of the "insertion method" of Lawson [55] has been implemented. For details see Witzgall et al. 2004 [86]. This method proceeds by inserting adaptively selected vertices, one at a time, into previously constructed TIN surfaces, so that from each insertion a new Delaunay triangulation and an associated TIN surface results. In this work, the data point to be inserted at each step is chosen to be the one that deviates the most from the current TIN surface. In the updated TIN surface, that deviation has then been reduced to zero as the point is now included in that surface.

The "adaptive" selection/insertion procedure typically stops before all points have been inserted. This allows intelligent thinning of large data sets, the post-triangulation adjustment of the resulting surface, and to take advantage of specifics of the selective process, as will be seen below. Most TIN methods, in use today, enter all data points into the TIN and delete data points subsequently as needed.

The adaptive selection procedure favors the early insertion of data points in areas of high variation of elevation over those in areas where elevations follow a planar pattern, that is, flat but not necessarily horizontal. In particular, objects hit by sideway scans exhibit strong variation of elevation in a narrow space. After about 30 % of the data points have been selected for insertion, objects stand out as starkly dense areas in a display of the triangulation (Figure 4.2).

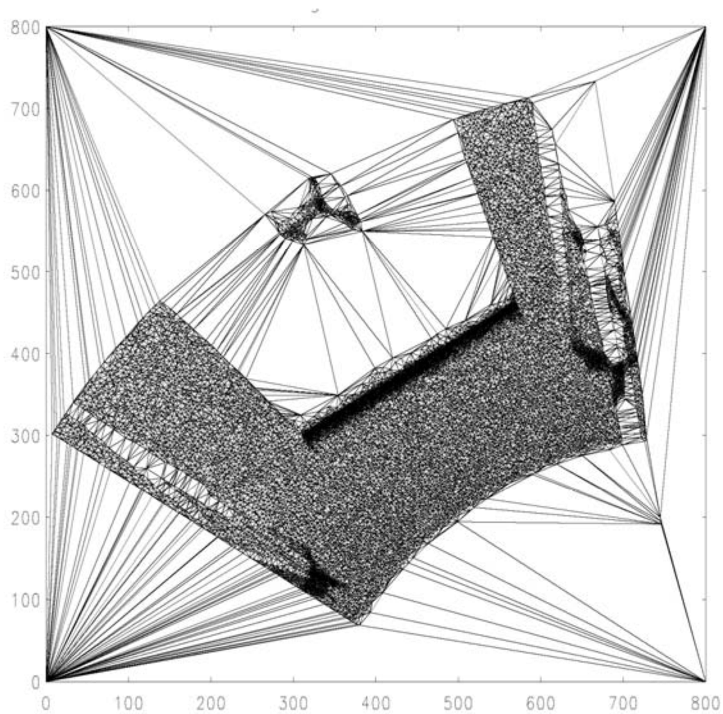


Figure 4.2. Triangulation of scene – Dense areas indicate objects.

The NIST version of the adaptive insertion method is specifically designed for the fastest possible retrieval of all data points whose footprints fall into any specified triangle of the triangulation. This feature facilitates the retrieval of object connected data points.

The goal of object segmentation is to identify and collect those data points in a point cloud that belong to an object of interest in the scene. In this work, point clouds were generated individually by single LADAR scans aimed at I-beams in various positions. The scan region included the I-beam and tripods supporting spheres intended as fiduciary devices. Object segmentation has two aspects:

- separate the data points garnered by the object from those referring to background
- distinguish between objects.

The program package TINsegment had been developed at NIST to accomplish such tasks. It proceeds in three steps.

Step 1 Apply the adaptive insertion method for creating a TIN with about 30 % of the data points selected.

This will produce a triangulation with heavy concentrations of small triangles, separated by large triangles (Figure 4.2).

Step 2 Delete triangles with an edge length exceeding a specified tolerance. The tolerance should be chosen to be a small multiple of the standard deviation in range measurements associated with the instrument used.

This will produce a set of "islands" of triangles connected by adjacency as shown in Figure 4.3.

Step 3 The data points corresponding to each island are collected.

This will produce an output file of data points of each island with separator lines between the data sets corresponding to different islands. The islands are represented by decreasing numbers of data points. An optional cutoff is provided to exclude small spurious islands. In this application, further analysis based on bounding boxes permits automatic recognition of the desired object, namely, the I-beam (see Section 4.3).

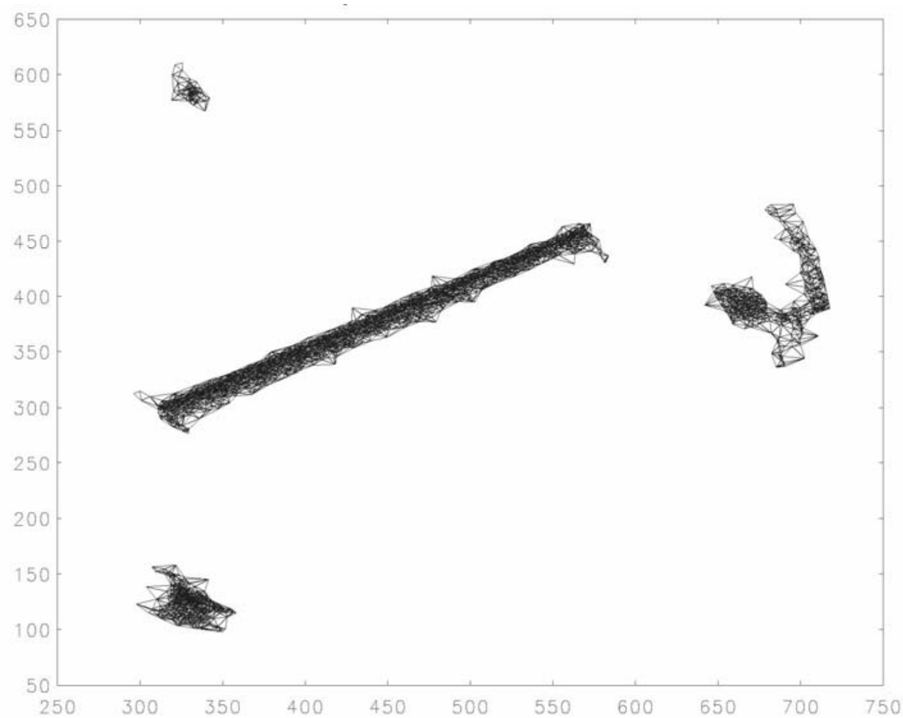


Figure 4.3. This figure shows the triangulation shown in Figure 4.2 after deletion of triangles of a specified edge length.

4.2 PRINCIPAL AXES

In order to determine the pose of a defined object, it is useful to identify a set of axes that represent the distribution of the voxel centers about the center of data mass. If an object is defined by a group of voxels, each with a center (x, y, z) associated with it, then a set of orthogonal axes relative to the center of data mass can be determined that extend in the directions of the longest axis of data, the second longest, and the third longest. These are called the *principal axes*. For a further discussion of this algorithm and its other applications in graphics see Lengyel [56]. Objects that are only linear or planar do not have a full set of principal axes and will not be considered as legitimate objects in this study. The principal axes are representative of the directions in which the data varies. To determine the principal axes assume that points $P_1, P_2, \ldots, P_N$ represent N points in 3D Euclidean space. We first calculate the mean position or center of data mass m (a vector) by

$$m = \frac{1}{N}\sum_{i=1}^{N} P_i \qquad (4.1)$$

We then construct a 3 x 3 matrix, since each P_i is 3D, called the covariance matrix

$$C = \frac{1}{N}\sum_{i=1}^{N}\left(P_i - m\right)\left(P_i - m\right)^T \qquad (4.2)$$

This covariance matrix is a symmetric matrix that represents the correlation between each pair of $P_i = (x_i, y_i, z_i)$ points. The natural axes of the set of points are determined as follows. First, the eigenvalues of C are found and ordered largest to smallest. The associated eigenvectors are then found. The eigenvector associated with the largest eigenvalue points in the direction of the points having the largest variation, generally the longest axis. The eigenvector associated with the next largest eigenvalue points in the direction of the next longest axis and similarly with the third. For the sake of the discussion of bounding boxes in the next section we will label the three principal axis eigenvectors as R, S, and T where R is the eigenvector associated with the largest eigenvalue, S with the second, and T with the third. They are centered at the center of data mass. Let these vectors be normalized to unit vectors. The process of developing the principal axes is related to the statistical technique called principal components analysis. For a discussion of this topic see Montgomery et al. [70].

4.3 BOUNDING BOX COMPARISON

Although there are uncertainties in locating the boundaries of LADAR scanned objects, it is possible to establish reference surfaces and vertices that can be used to compare against associated reference surfaces and vertices on the I-beams in the data base. Due to the statistical uncertainties of locating object edges, we use polygons to bound the object. One advantage of this approach is that if a polygon is also constructed around the ideal object from the database, such as an ideal I-beam, then comparisons can be made between well defined polygons. In the case of I-beams, polygons defined as bounding boxes are used. The approach is relatively straightforward. Every legitimate object is defined in terms of voxels and each of the voxels has a center (x, y, z) point. Once an object, in particular the I-beam, has been identified in terms of voxels then the vector from the LADAR scanner origin to each voxel center in the I-beam is projected onto the principal axes. The lengths of the different sides of the box enclosing all of the voxels defining the I-beam are given by the respective differences between the maximum and minimum values of the projections along each of the principal axes. The vertices of this box are then easily determined.

The construction of a bounding box given here is based on the construction given in Lengyel [56]. In this construction we start with all of the points from the point cloud that are identified as a significant object. Let these points be labeled $P_1, P_2, \cdots, P_N$, where N represents the number of point cloud points associated with the object. These labeled points are vectors from an origin. The projections of these vectors along the principle axes are given by the inner products of the points $P_1, P_2, \cdots, P_N$ with the principle axis vectors R, S, T defined above. These are direction vectors relative to the origin. The center Q of the bounding box is the point at which three planes lying halfway between the faces of the box meet. Let

$$a = \frac{\min_{1 \le i \le N}(P \bullet R) + \max_{1 \le i \le N}(P \bullet R)}{2},$$
$$b = \frac{\min_{1 \le i \le N}(P \bullet S) + \max_{1 \le i \le N}(P \bullet S)}{2},$$
$$c = \frac{\min_{1 \le i \le N}(P \bullet T) + \max_{1 \le i \le N}(P \bullet T)}{2},$$

where a, b, c are the average extents in the R, S, T directions. This has been implemented in the main object recognition scripts given in Appendix A.

Next, the specifications for an I-beam in the database can be used to construct the bounding box defining the I-beam. These points can also be used to generate the principal axes for the reference I-beam. Once these principal axes are known and the distance to the center of data mass of the measured I-beam from the scanner is known, it is then possible to construct the X-, Y-, and Z-rotations to transform the reference I-beam model and map it over the bounding box of the measured I-beam data. Then a sum of squared errors function can be used to compute the error between the associated vertices of the reference I-beam model and the measured I-beam bounding box. The segmented object will be associated with the known object which yields the smallest error. Figure 4.4 shows the bounding boxes for an identified object and the bounding box for an ideal object in the database.

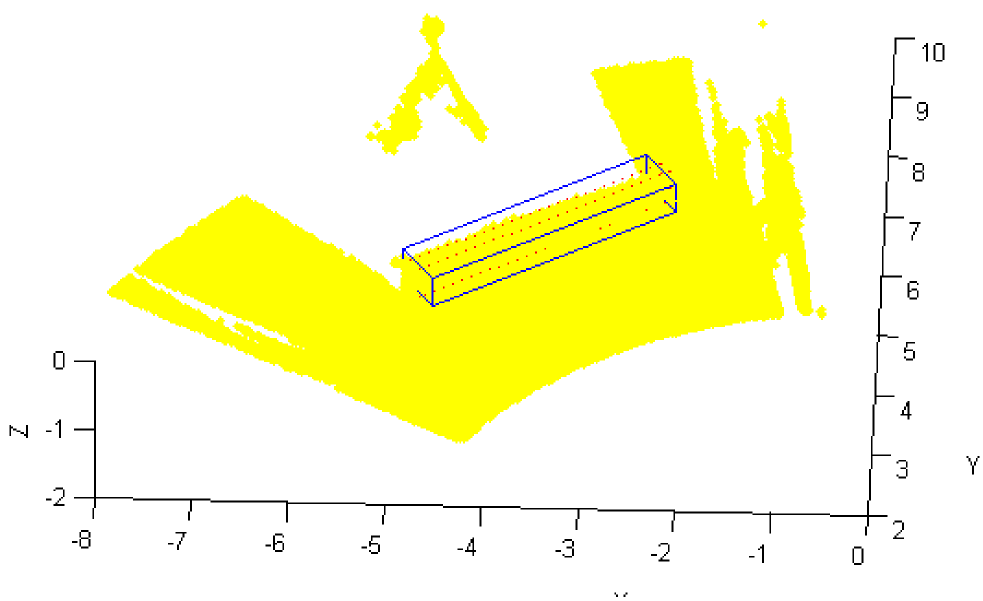


Figure 4.4. This figure shows a bounding box with solid lines around a potential object. It is compared with a model bounding box with dotted lines around it.

5. ESTABLISHING A REFERENCE FRAME

In order to determine the accuracy of the predicted pose of the I-beams, measurements at the four corner points of the upper flange of the I-beams were made using a Site Metrology System (SMS). This system is a laser-based system and the 3D coordinates, in the SMS frame of reference, are obtained by touching the tip of a digitizing tool to the point of interest. These coordinates were then converted into coordinates relative to the LADAR scanner frame through registration. Registration is the establishment of a rigid body transformation that brings the coordinate points in one frame into coordinate points in another frame. There is a large literature on the problems of registration. One of the major algorithms for 3D registration is called the Iterative Closest Point (ICP) algorithm by Paul Besl (see Besl and McKay [10]). This is a very general method that seeks a global matching of points or shapes. In our study, however, we use a technique of establishing three reference points in the scan view volume that are also measured by the SMS. These reference points and the measured location of the scanner relative to the SMS provide sufficient data to establish a rigid body transformation. In this chapter we describe the details of this process.

To measure the rotation of the I-beam, the four corners (a, b, c, and d in Figure 5.1) of the top flange were measured with the SMS that has a manufacturer stated accuracy of less than 2 mm. These measurements were taken every time the beam was moved. In an attempt to determine the "ease of use" of spheres for registration purposes, three spheres, nominal diameters of 152 mm (6 in), were located within the scene (see Figures 3.1, 4.1, 5.1, and 5.2). These spheres were used as additional targets for registration purposes, and several points on the surface of each sphere were measured using the SMS. These measurements and those obtained during the scans allow for the determination of the transformation matrix between the two coordinate frames, scanner and SMS.

Points on the scanner were also measured using the SMS to locate the scanner in the SMS coordinate frame. The scanner origin in the SMS coordinate frame was derived from these measurements as described in the next section.

To determine the transformation matrix, the initial intent was to use eight points, measured in both scanner coordinate frame and in SMS coordinate frame. These eight points were: the four corners of the upper flange (points a, b, c, d in Figure 5.1), the center of three spheres (A, B, and C) and the scanner origin. However, upon examining the LADAR data, it was very difficult to accurately isolate the corners of the upper flange in the point cloud due to noise and the phantom points or mixed pixels. The four corners could be determined visually; a method that was felt to be too subjective. However, four points could be used to determine the transformation matrix; they were the three sphere centers and the scanner origin. The reduction of usable control points highlights the need for redundancy of control points.

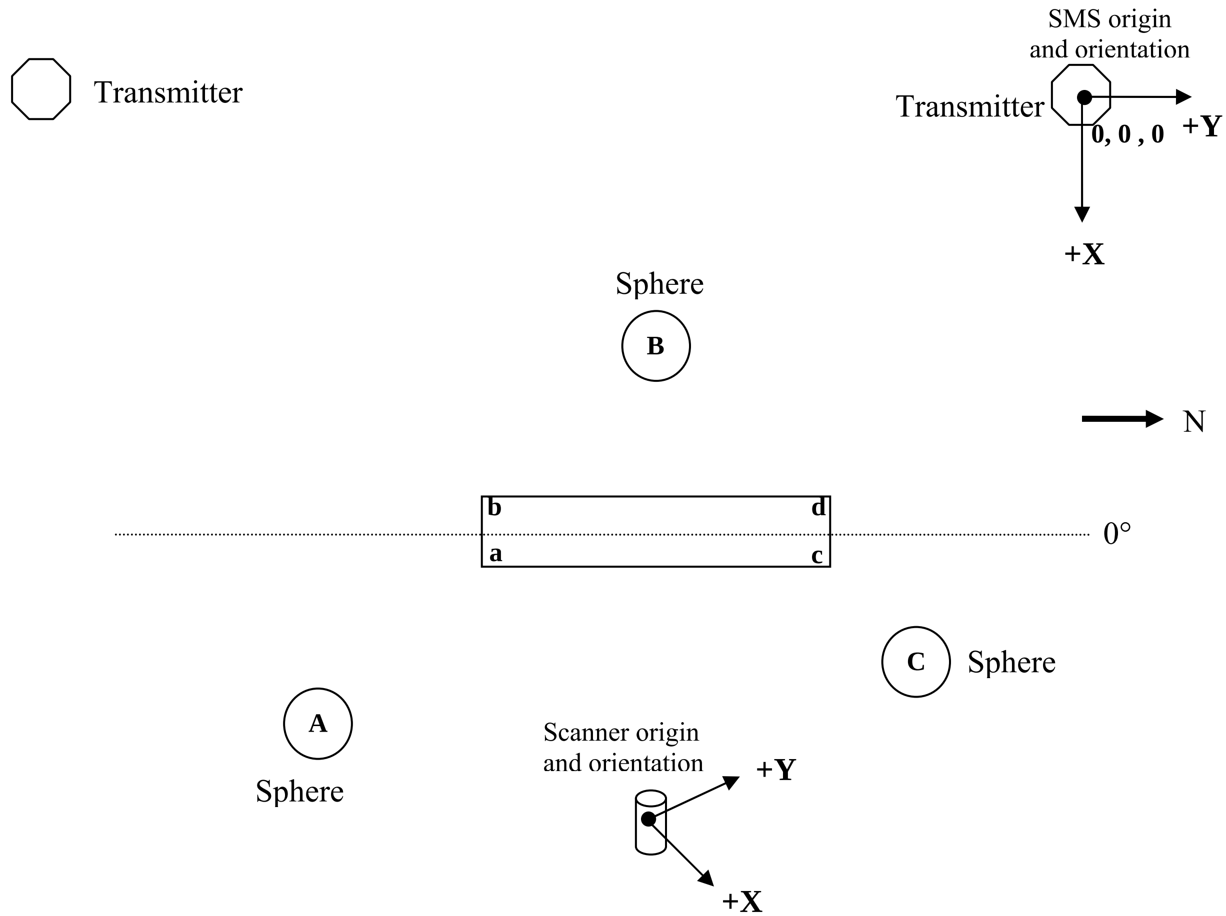


Figure 5.1. Schematic showing approximate orientation and centers of the SMS and scanner.

5.1 TARGET SPHERES

As alluded to above, the spheres were meant as secondary targets in the determination of the transformation matrix. The inclusion of the spheres was done in an effort to determine a potential sphere size for use in targeted registration and as an artifact for LADAR performance evaluation purposes. As such, the spheres used in these experiments were polystyrene spheres purchased from a hobby store and not machined spheres with tight tolerances on either the diameter and out-of-roundness or sphericity.

The diameters of the polystyrene spheres were measured using a micrometer with a resolution of 0.025 mm (0.001 in). However, since the spheres were not labeled A, B, or C when they were removed from the scene it was not possible to precisely identify measured sphere diameters with the original sphere located at positions A, B, and C in Figure 5.1. However Table 5.1 gives a list of 10 measurements made of the diameters of the three spheres designated as spheres 1, 2, 3, where the numbering does not indicate corresponding spheres to A, B, and C in Figure 5.1.

Table 5.1. Measured Sphere Diameter.

	Diameter (mm)		
Measurement #	Sphere 1	Sphere 2	Sphere 3
1	150.724	151.232	152.273
2	152.121	153.441	152.146
3	151.460	153.238	152.984
4	150.089	152.806	151.714
5	150.038	152.933	152.349
6	151.435	151.587	152.248
7	150.089	151.765	152.273
8	151.308	153.568	152.298
9	150.851	151.968	152.806
10	150.851	151.130	152.603
Average Diameter	150.896	152.367	152.370
Average Radius	75.448	76.183	76.185
Std. Dev. (Diameter)	0.695	0.932	0.356

5.2 DETERMINATION OF SCANNER ORIGIN

As there was no physical point on the instrument to designate the center or origin of the scanner, the scanner origin was derived by making several measurements. The scanner was mounted on a survey tripod and aligned over a point (P1) marked on the floor. This point on the floor was measured with the SMS system and yields the X- and Y- values for the scanner origin, in SMS coordinate frame. Four points around a circle on top of the scanner were also measured. A best-fit circle of these points yields an additional set of X- and Y- values of the scanner origin. The Z-value, in SMS coordinate frame, for the scanner was obtained by measuring a visually-located point on the scanner surface which corresponded to the rotation axis of the mirror. The measured values and derived scanner origin are given in Table 5.2.

Table 5.2. Instrument Origin in the SMS Coordinate Frame.

X and Y Values of Scanner Origin			
	Point #	X (m)	Y (m)
Coordinates of point, P1, on floor directly below instrument. Z-values are not relevant and not shown.	1	15.115	-1.519
	2	15.115	-1.519
	3	15.113	-1.520
	4	15.113	-1.520
	5	15.112	-1.521
Coordinates of 4 points on circle on top of instrument. Z-values are not relevant and not shown.	6	15.110	-1.551
	7	15.090	-1.520
	8	15.127	-1.518
	9	15.128	-1.518
Center of best fit circle of Pts. 6-9	10	15.109	-1.529
Average of Pts. 1-5, 10		15.113	-1.521
Std. Dev of Pts. 1-5, 10		0.002	0.004

Z-Values of Scanner Origin		
	Point #	Z (m)
Z-coordinate of point corresponding to rotation axis of mirror. X and Y coordinates are not relevant and not shown.	11	-0.128
	12	-0.128
	13	-0.128
	14	-0.128
	15	-0.129
Average of Pts. 11-15		-0.128
Std. Dev. of Pts. 11-15		0.0001

Instrument Origin in SMS Coordinate Frame: (15.113, -1.521, -0.128)
Note: Instrument Origin in Scanner Coordinate Frame is (0, 0, 0)

5.3 LOCATION OF SPHERE CENTERS: SCANNER COORDINATE FRAME

Three methods, A, B, and C, for fitting a sphere to a point cloud were examined. The spheres were 152 mm (6 in) in nominal diameter and made ofpolystyrene . Once a sphere fit had been achieved, the center of the sphere and its radius was determined. Of main interest, in the context of this work, was the center, as the registration of the instrument frame was directly based on these locations.

All methods examined here follow the general paradigm of “fitting” consisting of (i) specifying a parameter model of an object, and (ii) selecting the parameters of the model to minimize the model error for the point cloud as a whole. This involves the further choice of how to combine the individual errors into a single measure of deviation, such as the RMS (root-mean-square) measure for this purpose, defined as the square root of the mean of the squared model errors.

Two of the methods involved fitting a sphere where the radius of the sphere was constrained (Method A and Method C - fixed) to 76 mm (3 in) and the other method involved fitting a sphere where the radius was not constrained (Method B and Method C - free). It should be noted that all the data points, for each sphere, lie on the "side" of the sphere closest to the scanner and are not, as preferred, randomly scattered around the sphere – see Figure 5.2. This condition will likely adversely affect the goodness of the sphere fit.

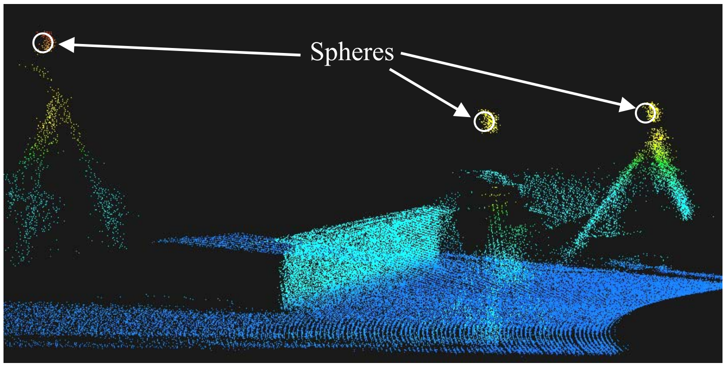


Figure 5.2. Data points located on side of sphere closest to scanner.

5.3.1 Method A

The procedure to determine the sphere center in Method A is based on a least squares fitting of a spherical equation with a constrained radius. Before the least squares method was applied, though, the 90 scan files of the I-beam and sphere scenes were preprocessed. In particular, the points associated with each sphere were isolated and saved to a file, i.e., one file contained the points for one sphere. This process yielded 270 files, but only 269 were used because one file became corrupted during the file writing process and could not be recovered. The method assumed that the radius of each of the spheres was 0.0762 m. It then performed a least squares fit based on

$$\min\left(\left(x-c_1\right)^2+\left(y-c_2\right)^2+\left(z-c_3\right)^2-0.0762^2\right)^2.$$

The scanned data values (x, y, z) were taken from files of measured data at different scan densities for each of the separate spheres, referred to as Sphere A, B, and C (Figure 5.1). The least squares process produced 269 estimates of the centers $\left(c_1, c_2, c_3\right)$ for the spheres.

The fitting process for each file was straightforward in that the points for each sphere were read and processed separately. Since the first two digits of all scanned points in a given file were the same, they were shifted out so as to relocate the origin of the data as near to $(0,0,0)$ as possible. This made all of the data used in the fitting the same order of magnitude thus lending to the numerical stability of the fitting algorithm. The sphere function and the shifted data points were provided to a nonlinear least squares software program along with an initial guess for the center. Instead of providing $(0,0,0)$ as the initial guess, each of the shifted data values was used as a

Table 5.3. Method A: Calculated Sphere Centers in Scanner Coordinate Frame.

Sphere A						
	Average			Std. Dev.		
Density	**x**	**y**	**z**	**x**	**y**	**z**
High	-4.787	3.302	-0.354	0.0168	0.0073	0.0067
Medium	-4.781	3.300	-0.354	0.0120	0.0069	0.0045
Low	-4.779	3.299	-0.358	0.0130	0.0068	0.0064
All Densities	-4.782	3.300	-0.355	0.0144	0.0071	0.0062
	Median					
High	-4.780	3.298	-0.351			
Medium	-4.779	3.298	-0.353			
Low	-4.776	3.298	-0.357			
All Densities	-4.779	3.298	-0.354			
Sphere B						
	Average			**Std. Dev.**		
Density	**x**	**y**	**z**	**x**	**y**	**z**
High	-4.727	7.867	-0.134	0.0059	0.0037	0.0031
Medium	-4.727	7.865	-0.136	0.0075	0.0050	0.0031
Low	-4.724	7.861	-0.140	0.0111	0.0101	0.0063
All Densities	-4.726	7.864	-0.137	0.0085	0.0072	0.0052
	Median					
High	-4.729	7.866	-0.133			
Medium	-4.729	7.865	-0.136			
Low	-4.728	7.862	-0.140			
All Densities	-4.729	7.865	-0.136			
Sphere C						
	Average			**Standard Deviation**		
Density	**x**	**y**	**z**	**x**	**y**	**z**
High	-1.078	5.843	-0.525	0.0042	0.0042	0.0009
Medium	-1.078	5.842	-0.529	0.0047	0.0057	0.0012
Low	-1.078	5.842	-0.533	0.0049	0.0097	0.0037
All densities	-1.078	5.842	-0.529	0.0046	0.0068	0.0038
	Median					
Density	**x**	**y**	**z**			
High	-1.079	5.843	-0.526			
Medium	-1.079	5.843	-0.529			
Low	-1.079	5.842	-0.533			
All Densities	-1.079	5.843	-0.529			

guess. This generated as many center estimates as there were data values in the given file. To each of these, the offset values were reintroduced to provide a list of potential sphere centers. The mean and the standard deviation of each of the estimated centers for the file were then computed. The final result after processing all of the data files was a list of means and standard deviations for the 269 scan files. The data was then grouped by scan density as shown in Table 5.3.

As described earlier, for any given data file, Method A computed *n* centers where *n* was the number of data points in that data file. The final "calculated" center was the average of these *n* centers and a corresponding standard deviation was also calculated. In Figures 5.3 to 5.5, the coordinates of the calculated centers for each data file are plotted. It can be seen in Figures 5.3 to 5.5 that the calculated centers for Sphere B have the least variability, especially for the high and medium density data sets. In Figures 5.4a-c, there are cases in which the error bars are not visible because they are hidden by the size of the plotted points. Plots of the *x*-, *y*-, and *z*-coordinates vs. density indicated an overall decreasing trend for the *z*-coordinate as the point density went from high to low. No consistent trends were noted for the *x*- or *y*-coordinates.

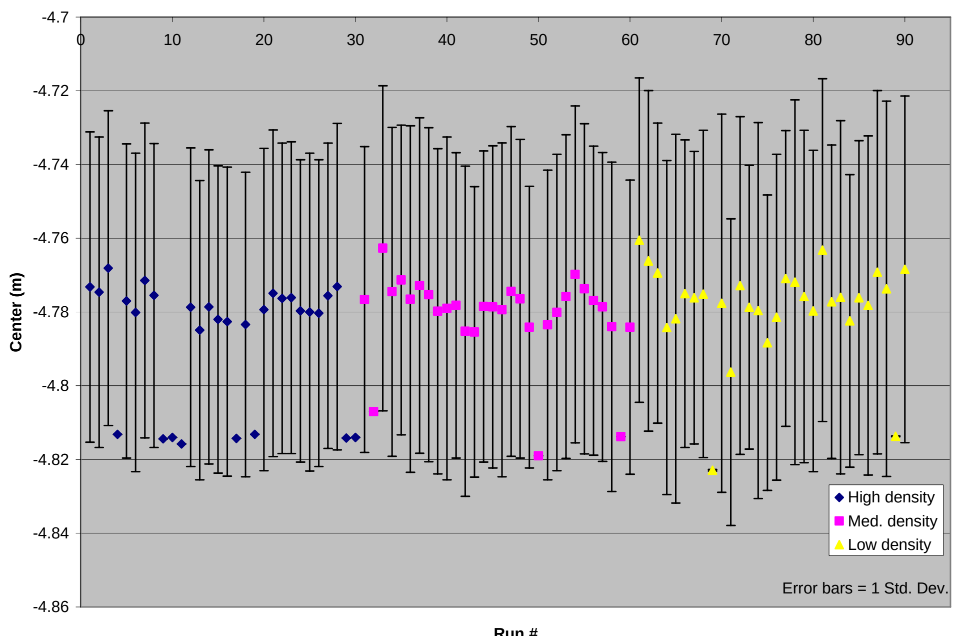


Figure 5.3a. The Average X-coordinate of Sphere A.

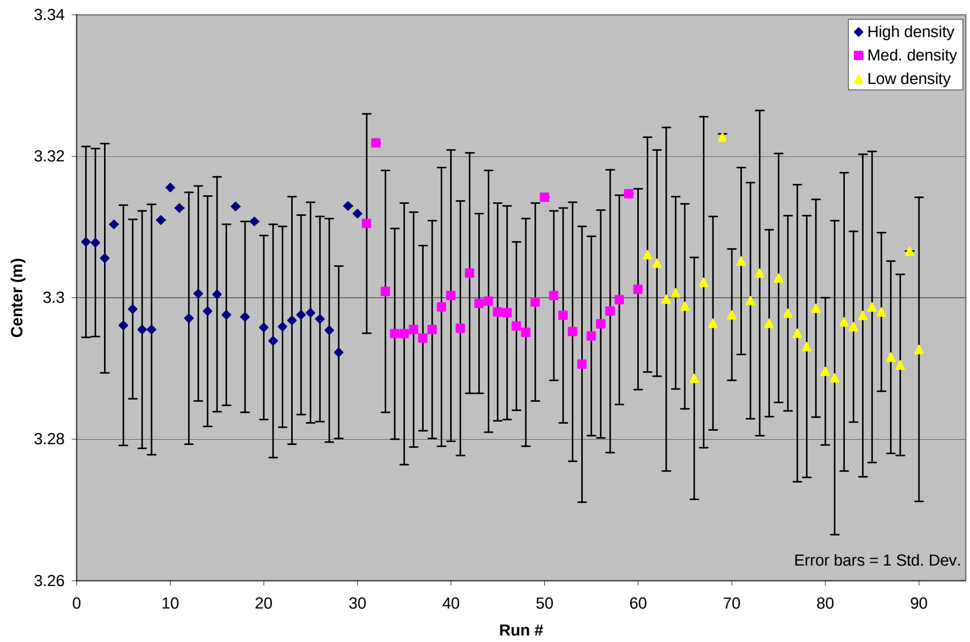


Figure 5.3b. The Average Y-coordinate of Sphere A.

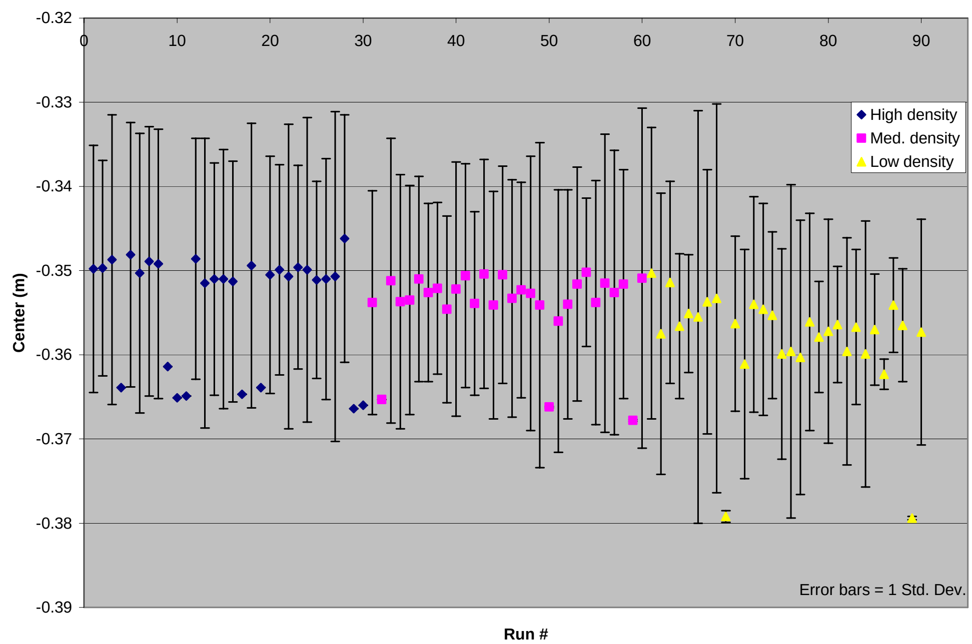


Figure 5.3c. The Average Z-coordinate of Sphere A.

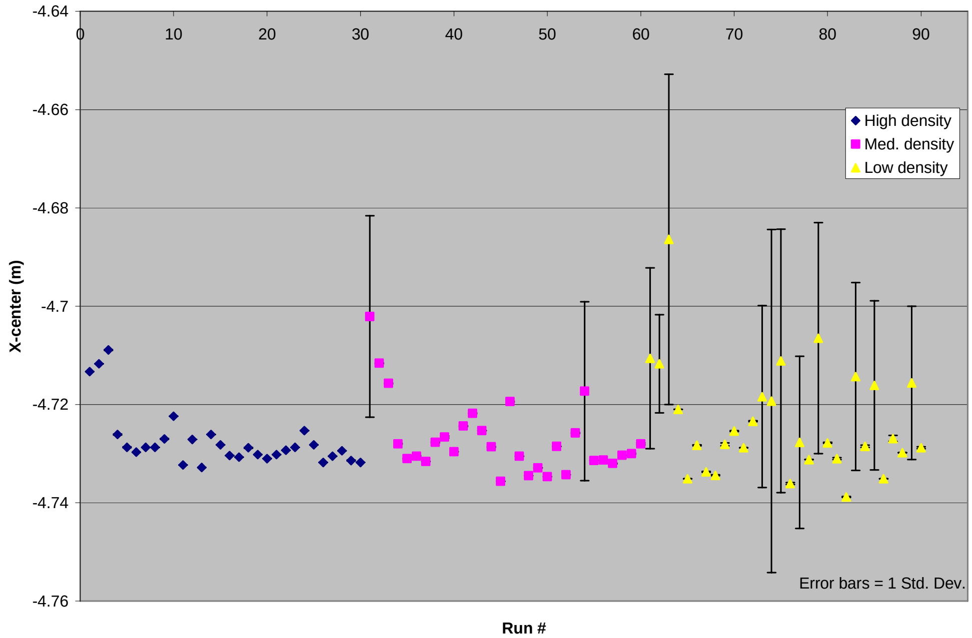


Figure 5.4a. The Average X-coordinate of Sphere B. Some error bars are not visible because they are hidden by the data point.

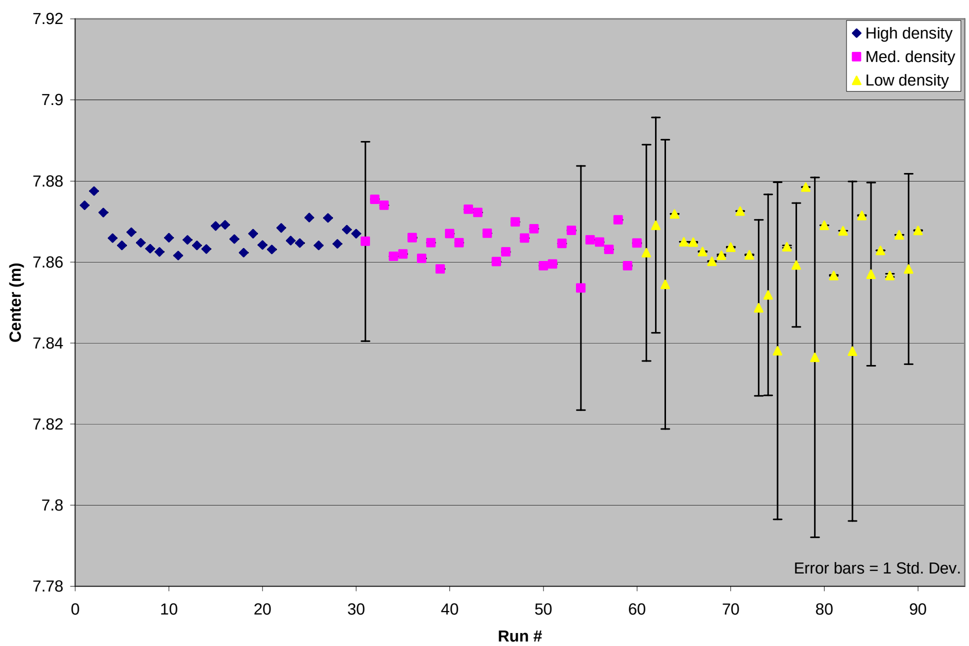


Figure 5.4b. The Average Y-coordinate of Sphere B. Some error bars are not visible because they are hidden by the data point.

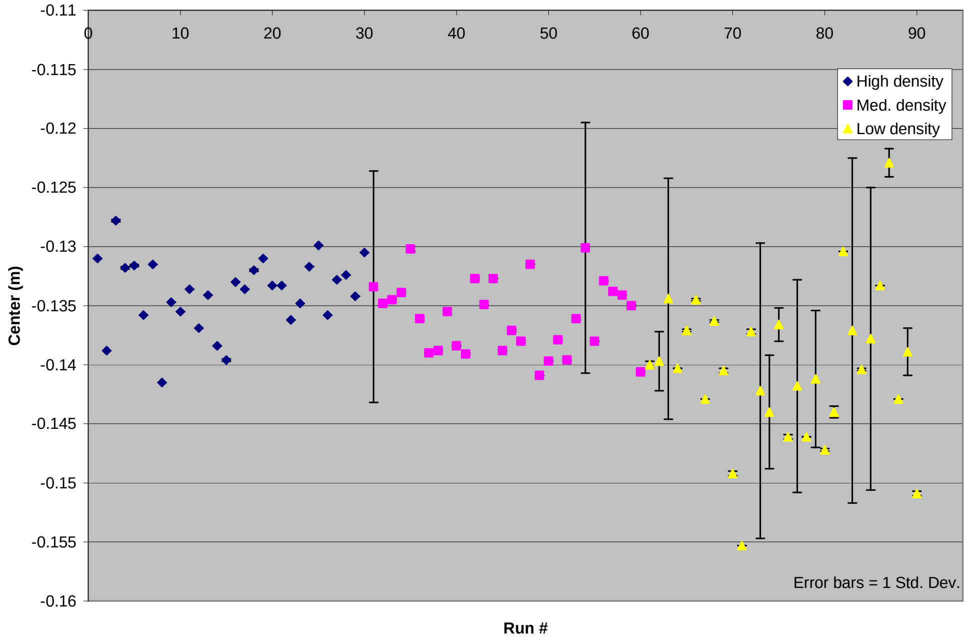


Figure 5.4c. The Average Z-coordinate of Sphere B. Some error bars are not visible because they are hidden by the data point.

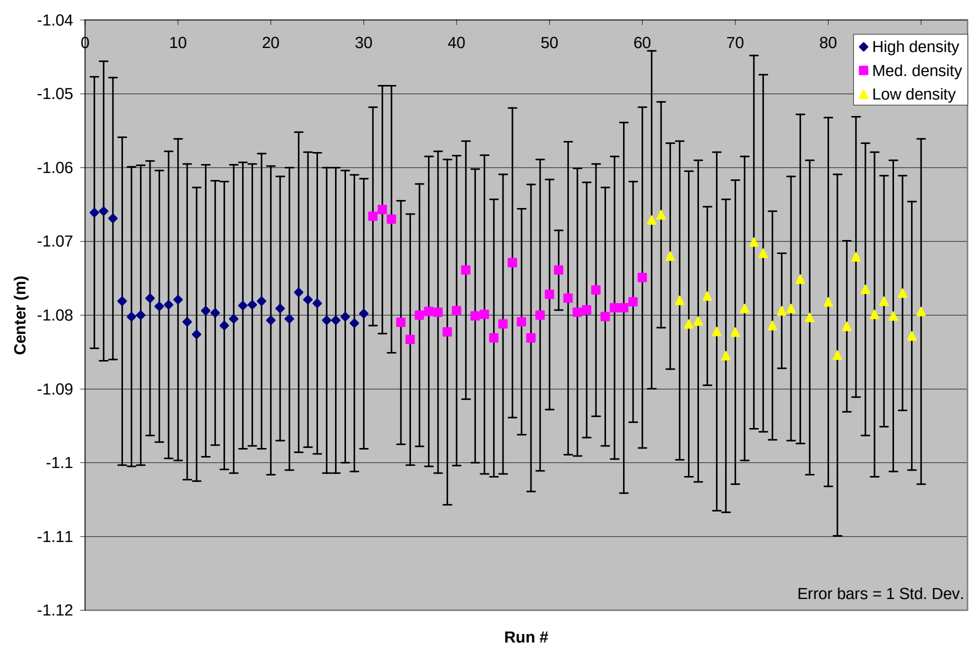


Figure 5.5a. The Average X-coordinate of Sphere C.

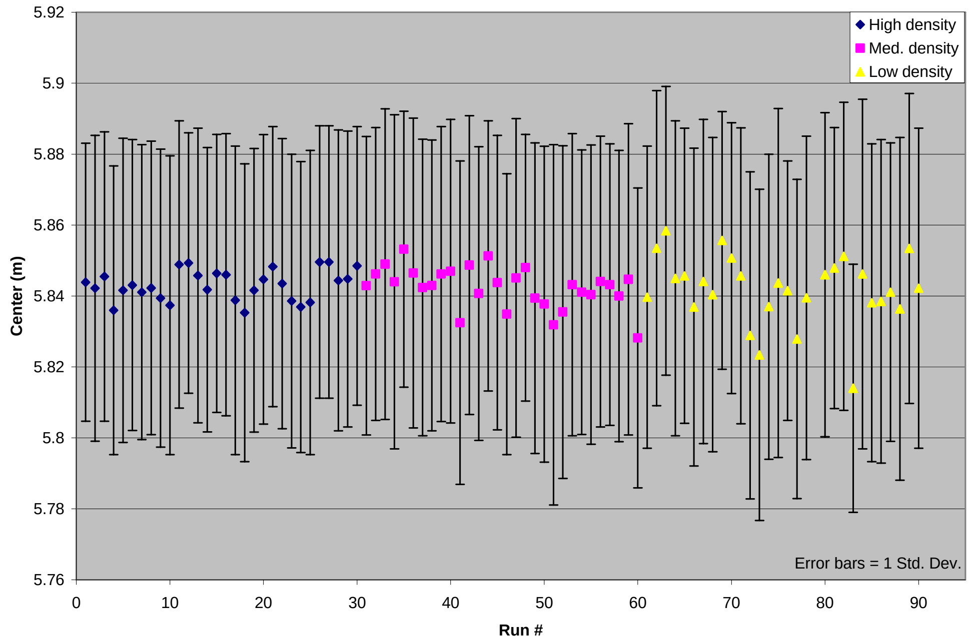


Figure 5.5b. The Average Y-coordinate of Sphere C.

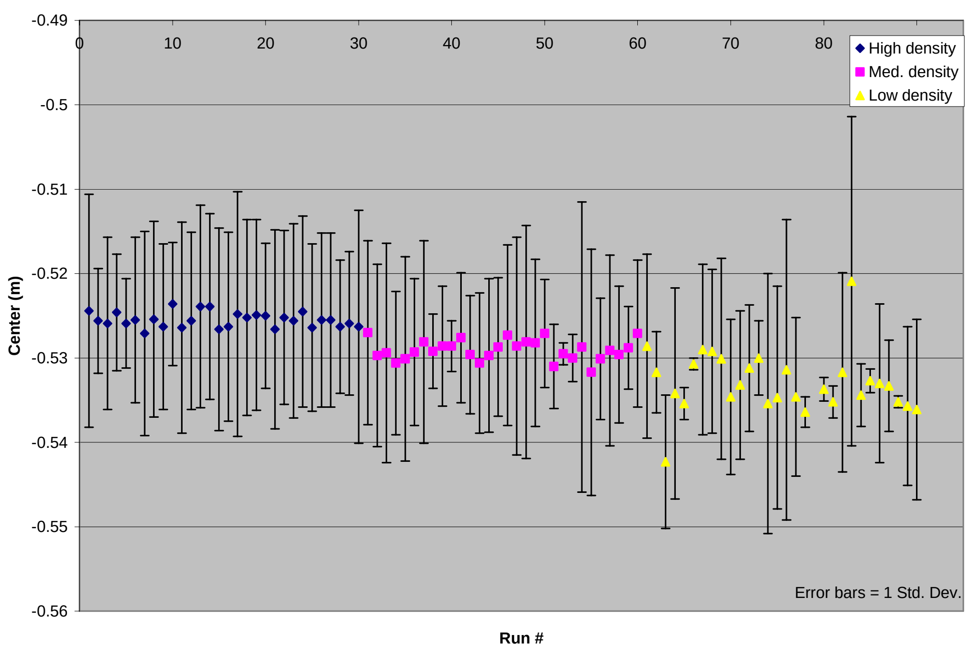


Figure 5.5c. The Average Z-coordinate of Sphere C.

5.3.2 Method B

This method does not fix the radius, but determines it based on the data. It is often referred to as an "algebraic" fit, because the sphere model chosen here is given by the equation below with parameters a_0, a_1, a_2, a_3:

$$x^2 + y^2 + z^2 + a_1 x + a_2 y + a_3 z + a_0 = 0, \qquad \text{(Eq. 5.1)}$$

the well-known equation for the sphere. Using the sum of squares to combine individual model errors, the optimization problem

$$\min \sum_i \left(x_i^2 + y_i^2 + z_i^2 + a_1 x_i + a_2 y_i + a_3 z_i + a_0 \right)^2$$

results. This is a linear regression problem for the parameters a_0, a_1, a_2, a_3. It is, therefore, readily solved. In this work the regression feature in MATLAB was used.

Equation 5.1 can be written as

$$\left(x + \frac{a_1}{2} \right)^2 + \left(y + \frac{a_2}{2} \right)^2 + \left(z + \frac{a_3}{2} \right)^2 - \frac{a_1{}^2}{4} - \frac{a_2{}^2}{4} - \frac{a_3{}^2}{4} + a_0 = 0.$$

This provides the center coordinates of the fitted sphere

$$x_0 = -\frac{a_1}{2},\ y_0 = -\frac{a_2}{2},\ z_0 = -\frac{a_3}{2}$$

and the radius

$$r = \sqrt{\frac{a_1^2}{4} + \frac{a_2^2}{4} + \frac{a_3^2}{4} - a_0}$$

This also points to a relationship between the model formulation in Method A and the algebraic model formulation used here. Only the parameters differ. Note that the parameters chosen here, while enabling regression, preclude fixing the radius.

The results show that almost all of the calculated radii were under or overestimated. The computed radii ranged from 49 mm to 109 mm as compared to a nominal radius of 76 mm. The average of the calculated radii for Sphere B came closest to the nominal value. Those for Sphere A and C did not with the average radius for Sphere A closer to the nominal value than that for Sphere C.

For each sphere, there is a maximum of 90 possible data sets. The radius was used to filter out "bad" sphere fits. The criterion used was to eliminate data sets when the calculated radius did not fall between 76 mm ± 5 mm. However, when this criterion was used, only four (4) data sets were valid for Sphere A, 30 for Sphere B, and none for Sphere C.

To increase the number of "valid" sets for Spheres A and C, the criterion was changed. For Sphere A, data were kept when the calculated radius was within 76 mm ± 10 mm. For Sphere C, the calculated radii, for all 90 data sets, ranged from 50 mm to 56 mm, and data were kept when the calculated radius was greater than 55 mm. The averages and standard deviations of the radii and sphere centers are given in Table 5.4 and shown in Figure 5.6. Due to the limited number of "valid" data sets, the data in Table 5.4 are not sub-divided by densities as is possible in Method A.

Table 5.4. Method B: Calculated Sphere Centers in Scanner Coordinate Frame.

		x (m)	y (m)	z (m)	Radius (m)
Sphere A					
All data sets (n = 90)	Average	-4.781	3.299	-0.356	0.0562
	Std. Dev.	0.015	0.007	0.007	0.0090
Filter by radius, r: r = 76 mm ± 10 mm (n = 9)	Average	-4.807	3.311	-0.365	0.0714
	Std. Dev.	0.006	0.005	0.005	0.0046
Sphere B					
All data sets (n = 90)	Average	-4.721	7.858	-0.136	0.0688
	Std. Dev.	0.009	0.009	0.005	0.0059
Filter by radius, r: r = 76 mm ± 5 mm (n = 30)	Average	-4.729	7.865	-0.137	0.0745
	Std. Dev.	0.005	0.006	0.005	0.0028
Sphere C					
All data sets (n = 90)	Average	-1.118	5.812	-0.527	0.0536
	Std. Dev.	0.390	0.268	0.019	0.0012
Filter by radius, r: r > 55 mm (n = 17)	Average	-1.077	5.843	-0.528	0.0551
	Std. Dev.	0.005	0.007	0.004	0.0003

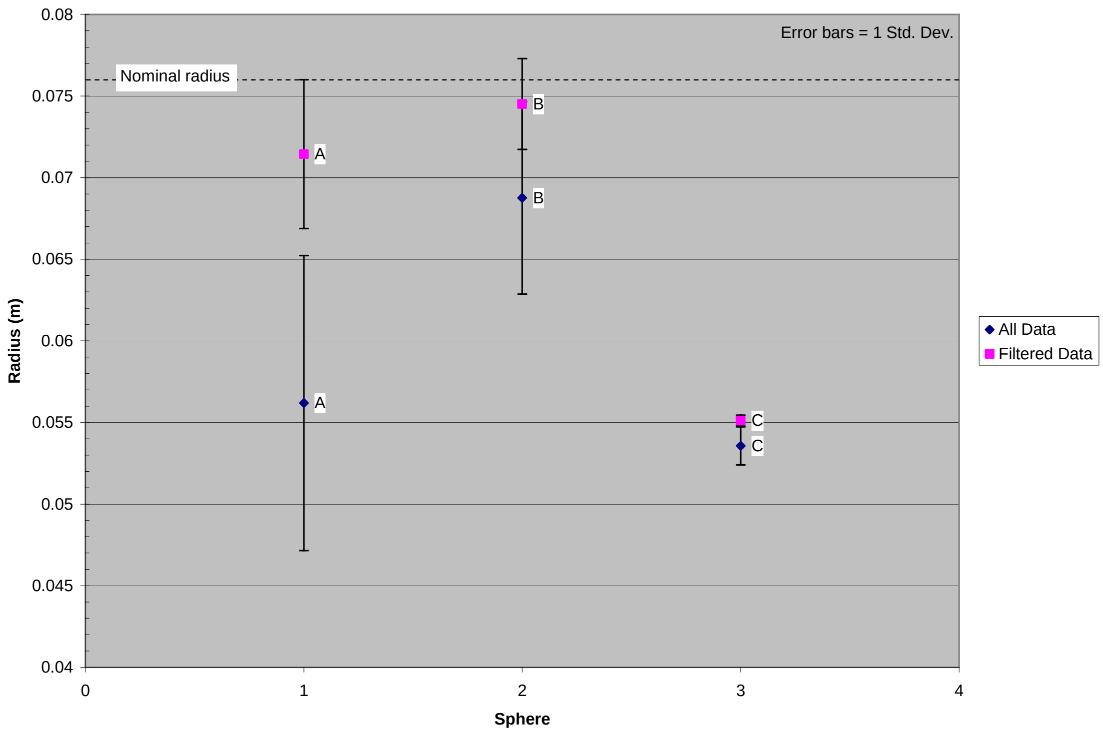


Figure 5.6. Method B: Calculated Sphere Centers.

In an effort to further understand why the calculated radius differed from the nominal radius, a plot of the deviations of the radius was made. The radius residual was the difference between the distance from the calculated center to the data point and the calculated radius or

$$radius \quad residual = \sqrt{\left(x_i - x_{0\_calc}\right)^2 + \left(y_i - y_{0\_calc}\right)^2 + \left(z_i - z_{0\_calc}\right)^2} - r_{calc} \, .$$

As Method B minimizes the deviations from Eq. 5.1, the error of the algebraic fit was determined. The algebraic residual is

$$algebraic\ residual = \left(x_i^2 + y_i^2 + z_i^2\right) - \left(a_1 x_i + a_2 y_i + a_3 z_i + a_0\right).$$

The plots of the radius residuals and the algebraic error for a data file, where the calculated radius was 54 mm, are shown in Figure 5.7. Two things stand out in Figure 5.7. The first being the scalloped pattern for the algebraic residual. The reason for this regular pattern is not known and requires further examination. The second being that the radius residuals are much larger than the algebraic residuals. This is expected as Method B minimizes the algebraic fit and not the geometric fit, that is,

$$Algebraic\ Fit: \quad Min\left[(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2 - r^2\right]$$

$$Geometric\ Fit, minimizes\ the\ orthogonal\ distance: \quad Min\left[\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} - r\right]^2$$

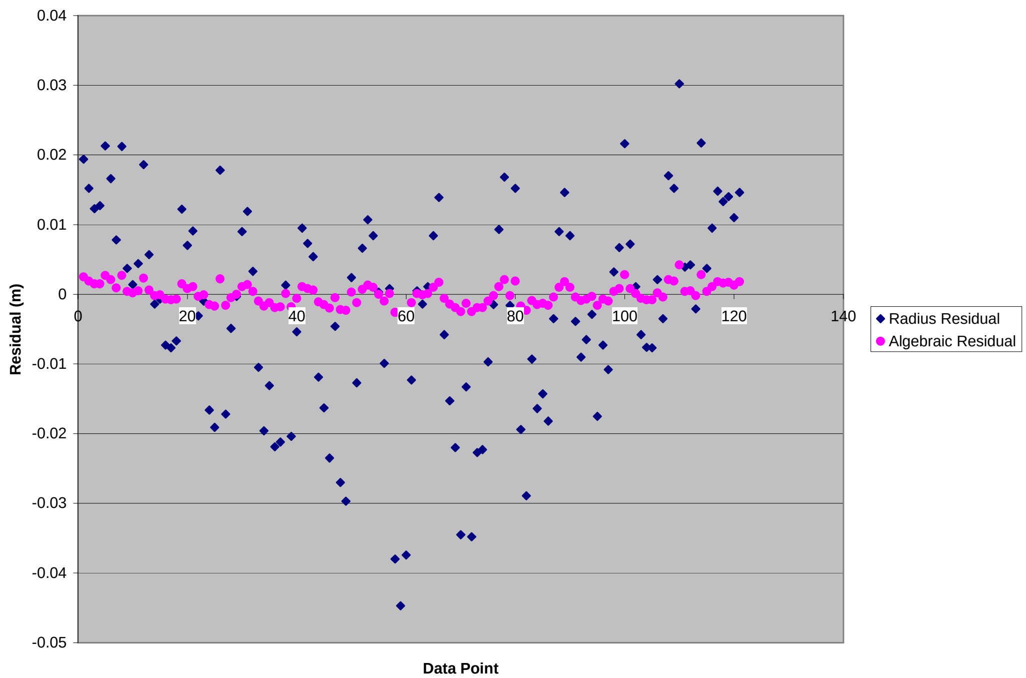


Figure 5.7. Plot of Residuals.

5.3.3 Method C

A third method was used to determine the sphere centers. A commercially available software package, Spatial Analyzer, was used to compute the sphere centers and radii. The software allows the user to select if the radius were constrained or unconstrained. Of the 270 sphere data files, the sphere centers and radii were computed for only 192 (64, of a possible 90, each for spheres A, B, and C) of these. Additionally, the software allows the user to ignore data that is greater than a specified fit tolerance. For this method, the tolerance selected was 50 cm which is the manufacturer specified worst case range uncertainty of the LADAR. In almost all of the cases, all of the data points were within the tolerance and were used. Inclusion of the points that were out-of-tolerance did not significantly change the calculated sphere center or radius.

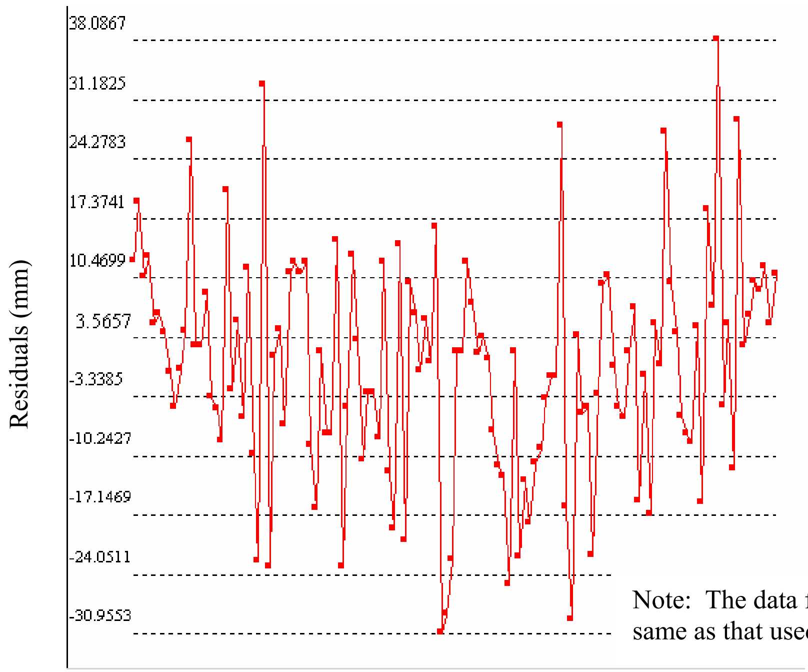
Unlike Method B which consistently resulted in a calculated sphere radius of about 56 mm for Sphere C, the calculated radii for Sphere C were not consistently underestimated, and they ranged in value from 49 mm to 95 mm. The results are summarized in the Table 5.5. As in

Method B, only centers where the radius was 76 mm ± 10 mm were included in the calculation of the median and average values given in Table 5.5. For Spheres A, 15 of the 64 radii met this criterion. For Sphere B, 57 of the 64 radii met this criterion and for Sphere C, 15 of the 64 radii met this criterion. The calculated radius for Sphere B came closest to the nominal value, and the larger number of "valid" data sets for Sphere B seems to suggest more confidence in the results obtained for Sphere B than for Spheres A or C. This was also observed in Method B.

Table 5.5. Method C: Calculated Sphere Centers and Radii.

		Unconstrained				**Constrained (radius = 0.0762 m)**		
		x (m)	**y (m)**	**z (m)**	**Radius (m)**	**x (m)**	**y (m)**	**z (m)**
Sphere A	Median	-4.813	3.312	-0.362	0.071	-4.818	3.313	-0.362
	Average	-4.814	3.313	-0.364	0.072	-4.796	3.306	-0.358
	Std. Dev	0.008	0.004	0.006	0.005	0.043	0.018	0.014
Sphere B	Median	-4.731	7.870	-0.135	0.075	-4.732	7.870	-0.136
	Average	-4.730	7.870	-0.136	0.076	-4.730	7.871	-0.136
	Std. Dev.	0.008	0.006	0.005	0.004	0.006	0.004	0.005
Sphere C	Median	-1.091	5.876	-0.535	0.070	-1.096	5.883	-0.535
	Average	-1.089	5.870	-0.534	0.069	-1.086	5.861	-0.532
	Std. Dev	0.009	0.024	0.008	0.003	0.019	0.043	0.009

After the software determines the best fit, a plot of the radius residuals is displayed and is shown in Figure 5.8. As seen in Figure 5.8, these residuals have the same pattern as the radius residuals in Figure 5.7.



Note: The data file used is the same as that used in Figure 5.7.

Figure 5.8. Radius Residuals for Method C - Free.

5.4 SUMMARY: SPHERE CENTERS IN SCANNER COORDINATE FRAME

The sphere centers and radii computed using the methods described in the previous sections are summarized in Table 5.6. In Table 5.6, the terms “Fixed” and “Free” are used to indicate if the radius was constrained or unconstrained in the various methods.

A review of the table shows that the values for Sphere B are the most consistent for all methods as noted in the observations made for the individual methods. The reason for the better results for Sphere B is not clear as the effects of angle of incidence and point of view of target from scanner is minimized by the choice of using spheres as targets. One possible reason for the better results for Sphere B could be due to the distance of Sphere B from the scanner. Both Spheres A and C are located approximately 5 m from the scanner while Sphere B is about 9 m from the scanner. The longer distance to Sphere B would result in a larger region of coverage of the sphere surface and thus a better distribution of the points on the sphere. On the other hand, the sphere that is farther away from the scanner would yield a lesser number of measurements than one that was closer. It appears that in this case, the benefits of better point distribution out-weighed benefits of more points. The optimal target distance depends on the size of the sphere and the selected point density of the instrument; the determination of which is not within the scope of this project.

Included in Table 5.6 is a column labeled RMS. The RMS error was defined as:

$$RMS = \sqrt{\frac{\sum_{i=1}^{n}\left(\sqrt{(x_i - x_{0\_calc})^2 + (y_i - y_{0\_calc})^2 + (z_i - z_{0\_calc})^2} - r_{calc}\right)^2}{n}}$$

where x_{0_calc}, y_{0_calc}, z_{0_calc} is the calculated sphere center and r_{calc} is the calculated radius. As seen in Table 5.6, the RMS value is not a good gage of how well the radius was calculated. For Sphere C, Method B - free ($r_{calc.}$ = 55 mm vs. $r_{nominal}$ = 76 mm), the RMS value was equal to or lower than those for Methods A, B and Method C – fixed even though these methods yielded calculated radii closer to the nominal value.

As these spheres are used to determine the transformation matrix between the scanner and the SMS coordinate frame, it is not possible to check (i.e., compare transformed sphere centers to SMS values for the sphere centers) which of the methods, A, B, or C, yielded the best results. A more rigorous test procedure involving well characterized spheres (known diameter and sphericity), a high resolution LADAR and a laser tracker would be necessary to evaluate these methods and is beyond the scope of this project. The authors would like to comment that the algorithm verification procedure and the procedure to determine the optimal distance of the sphere are ideal tasks for a LADAR performance evaluation facility. For this series of experiments, the coordinates of the sphere centers used to determine the transformation matrix were set equal to the average of all the median values obtained from Methods A, B, C and are given at the end of Table 5.6. The median values were used as they are less susceptible to outliers.

Table 5.6. Comparison of Sphere Centers and Radii for All Methods in Scanner Coordinate Frame.

	Median Values					**Average Values**				
	x (m)	**y (m)**	**z (m)**	**RMS (m)**	**Radius (m)**	**x (m)**	**y (m)**	**z (m)**	**RMS (m)**	**Radius (m)**
Sphere A										
Method B - free	-4.805	3.311	-0.364	0.015	0.070	-4.807	3.311	-0.365	0.016	0.071
Method C - free	-4.813	3.312	-0.362	0.015	0.071	-4.814	3.313	-0.364	0.015	0.072
Method A - fixed	-4.779	3.298	-0.354	0.029	0.076	-4.782	3.300	-0.355	0.028	0.076
Method C - fixed	-4.818	3.313	-0.362	0.015	0.076	-4.796	3.306	-0.358	0.022	0.076
Avg – All	-4.804	3.308	-0.360	0.019	0.073	-4.800	3.307	-0.361	0.020	0.074
Avg - Fixed	-4.798	3.306	-0.358	0.022	0.076	-4.789	3.303	-0.357	0.025	0.076
Avg - Free	-4.809	3.311	-0.363	0.015	0.070	-4.811	3.312	-0.365	0.016	0.072
Std. Dev - All	0.018	0.007	0.005	0.007	0.003	0.014	0.006	0.005	0.006	0.002
Std. Dev - Fixed	0.028	0.011	0.006	0.010	0.000	0.010	0.004	0.002	0.004	0.000
Std. Dev - Free	0.006	0.000	0.001	0.000	0.001	0.005	0.001	0.001	0.001	0.001
Sphere B										
Method B - free	-4.729	7.863	-0.136	0.015	0.074	-4.728	7.865	-0.137	0.015	0.075
Method C - free	-4.731	7.870	-0.135	0.014	0.075	-4.730	7.870	-0.136	0.014	0.076
Method A - fixed	-4.729	7.865	-0.136	0.015	0.076	-4.726	7.864	-0.137	0.015	0.076

	Median Values					Average Values				
	x (m)	y (m)	z (m)	RMS (m)	Radius (m)	x (m)	y (m)	z (m)	RMS (m)	Radius (m)
Method C - fixed	-4.732	7.870	-0.136	0.014	0.076	-4.730	7.871	-0.136	0.014	0.076
Avg – All	-4.730	7.867	-0.136	0.015	0.075	-4.729	7.867	-0.136	0.015	0.076
Avg - Fixed	-4.730	7.867	-0.136	0.015	0.076	-4.728	7.867	-0.136	0.015	0.076
Avg - Free	-4.730	7.866	-0.136	0.015	0.075	-4.729	7.867	-0.136	0.015	0.075
Std. Dev - All	0.001	0.003	0.000	0.001	0.001	0.002	0.003	0.000	0.001	0.001
Std. Dev - Fixed	0.002	0.003	0.000	0.001	0.000	0.003	0.005	0.000	0.001	0.000
Std. Dev - Free	0.001	0.005	0.000	0.001	0.001	0.001	0.004	0.001	0.001	0.001
Sphere C										
Method B - free	-1.080	5.845	-0.526	0.016	0.055	-1.077	5.843	-0.528	0.016	0.055
Method C - free	-1.091	5.876	-0.535	0.015	0.070	-1.089	5.870	-0.534	0.016	0.069
Method A - fixed	-1.079	5.843	-0.529	0.029	0.076	-1.078	5.842	-0.529	0.029	0.076
Method C - fixed	-1.096	5.883	-0.535	0.016	0.076	-1.086	5.861	-0.532	0.022	0.076
Avg – All	-1.087	5.862	-0.531	0.019	0.069	-1.083	5.854	-0.531	0.021	0.069
Avg - Fixed	-1.088	5.863	-0.532	0.023	0.076	-1.082	5.852	-0.531	0.026	0.076
Avg - Free	-1.086	5.860	-0.530	0.016	0.062	-1.083	5.857	-0.531	0.016	0.062
Std. Dev - All	0.008	0.021	0.004	0.007	0.010	0.006	0.013	0.003	0.006	0.010
Std. Dev - Fixed	0.012	0.028	0.004	0.009	0.000	0.005	0.013	0.002	0.005	0.000
Std. Dev - Free	0.008	0.022	0.006	0.001	0.010	0.008	0.019	0.004	0.000	0.010
Coordinates used for Sphere Centers = Median values, Average All.										
Sphere A	**-4.804**	**3.308**	**-0.360**							
Sphere B	**-4.730**	**7.867**	**-0.136**							
Sphere C	**-1.087**	**5.862**	**-0.531**							

5.5 FINAL TRANSFORMATION

Using the estimated sphere centers as reference points to transform the SMS coordinates into the scanner frame is an application of range image registration. There is a substantial literature on this topic and only a few references are given here: Haralick et al. [32], Herbin et al. [36], Besl and McKay [10], Turk and Levoy [78], Monga and Benayoun [69], Bergevin et al. [7], Hsieh et al. [40], Soucy and Ferrie [77], Johnson [45], Cunnington and Stoddart [17], Curless [18], Bernardini and Rushmeier [9], Hemayed et al. [35], Williams and Bennamoun [84], Laboureux and Häusler [53], Lucchese et al. [62].

As described earlier in Section 5 the location and pose of the I-beam were determined in the SMS coordinate system during the preparatory phase of the experiment. This information was used to assess the accuracy of determining the location and pose by means of LADAR scanning.

Therefore, the SMS coordinates needed to be registered, that is aligned with the scanner coordinates by a coordinate transformation, *T*, consisting of a rotation and a translation.

To this end, three spheres were placed in the vicinity of the I-beam and their centers *A*, *B*, *C* determined in the SMS coordinate frame. Similarly, scanning information was used to find the scanner coordinates *A'*, *B'*, *C'* of the sphere centers. The coordinate transformation *T* was then selected so as to bring the pairs of target points into their best possible alignment as measured by the sum of squares of the distances between corresponding coordinates:

$$\min_T \left[\|A' - T(A)\|^2 + \|B' - T(B)\|^2 + \|C' - T(C)\|^2 \right]$$

A solution to this minimization problem – sometimes called the "Procrustes problem – can be found in the text by Golub and Van Loan [28], Chapter 12, Section 4.1 (also O'Leary [73]). In-house developmental software, called "pointreg" produced the transformation.

$$T : \text{Scanner coords} = \text{Rotation*(SMS coords - E)} + \text{E'}$$

where

$$\text{E} = \frac{\text{A+B+C}}{3} = (10.1345, -2.0517, -0.3965)$$

$$\text{E'} = \frac{\text{A'+B'+C'}}{3} = (-2.6553, 4.2593, -0.2567)$$

are the centroids of the respective target points and

$$\text{Rotation} = \begin{bmatrix} 0.4363 & 0.8998 & 0.0026 \\ -0.8998 & 0.4363 & -0.0012 \\ -0.0022 & -0.0018 & 0.9999 \end{bmatrix}$$

For information about "pointreg" and related issues consult Lawrence and Witzgall [54].

The locations of the spheres in the SMS coordinate frame were then computed using the transformation based upon Table 5.6. The values are given in Table 5.7.

Table 5.7. SMS Coordinate Frame: Sphere Centers and Radii.

	x (m)	y (m)	z (m)	Radius (m)
Sphere A				
Method A – fixed	10.073	-4.382	-0.498	0.076
Method B – free	10.052	-4.396	-0.505	0.083
Method C – free	10.053	-4.396	-0.505	0.083
Method C – fixed	10.054	-4.395	-0.505	0.076
Average	10.058	-4.392	-0.503	0.079
Std. Dev.	0.010	0.007	0.004	0.004
Sphere B				
Method A – fixed	5.985	-2.345	-0.285	0.076
Method B – free	5.995	-2.341	-0.285	0.079
Method C – free	5.995	-2.341	-0.285	0.079
Method C – fixed	5.995	-2.342	-0.286	0.076
Average	5.993	-2.342	-0.285	0.078
Std. Dev.	0.005	0.002	0.001	0.002
Sphere C				
Method A – fixed	9.362	0.076	-0.673	0.076
Method B – free	9.377	0.051	-0.668	0.078
Method C – free	9.377	0.051	-0.668	0.078
Method C – fixed	9.376	0.051	-0.667	0.076
Average	9.373	0.057	-0.669	0.077
Std. Dev.	0.007	0.013	0.003	0.001

6. RESULTS

This chapter summarizes the results of the pose determination and the associated errors for the experimental study.

6.1 COMPARISON OF SEGMENTATION METHODS

In their current state, both of the binning and TIN methods of segmentation have benefits as well as some difficulties. In the case of binning, there is a single program that runs the segmentation and identification procedures. Currently, the TIN method requires execution of two separate processes: data segmentation and object identification. In future work, these tasks could be combined. Additionally, the binning program has built in graphics display, whereas the TIN method requires a separate graphics display program. Once the data is segmented, either by binning or TINning, the object pose is determined using the method of principal components analysis for determining the principal axes, and the object is identified using the bounding box construction.

In the binning program, object segmentation is done in two steps. In the first step, all potential objects are segmented and in the second step, the groupings of voxels for each potential object are analyzed to determine whether there are three axes present. For example, if a potential object is determined to be a concatenated vertical string of voxels, then that potential object would be dismissed as not satisfying the three axes criterion. In the TIN method, potential objects are determined by the density and size of triangles. The capability of eliminating 2D objects based on the three axes criterion is currently not incorporated but can be easily implemented. Potential objects appear as groups of small triangles. Figure 4.3 shows the triangulation after triangles with sides larger than a specified length are eliminated.

In both methods, it is fairly obvious, visually, where the I-beam is located, but the real challenge is to have it recognized without human intervention, have its pose, relative to the scanner, calculated, and have these tasks done in real-time.

6.2 ERRORS IN OBJECT ID AND POSE

A total of 90 scan files were obtained. The files covered three scan densities for each angular position of the two beams identified as I-Beam A and B. For each scan density three repetition scans were made to estimate scan variability. The angular positions on a floor axis system were set at 0^o, 30^o, 45^o, 60^o, 90^o. We will refer to these as floor angles, and these are not expected to be the same as the calculated orientation in the scanner coordinate frame. However, the relative differences between the floor angles and the calculated angles should be similar. The measured rotations of the I-beams, based on the SMS data, are given in Table 6.1. In the tables in this chapter, the first letter of the file names indicates the I-beam that was scanned: A or B. The units for the coordinates in these tables are meters and angles are measured in degrees relative to the scanner coordinate system.

Table 6.1. Orientations of the two I-beams as measured using the SMS data in the scanner coordinate frame.

File Prefix	SMS Ref. Orientation (º)	Angle Relative to the 0º position
A0	32.161	0
A30	62.186	30.025
A45	77.725	45.564
A60	92.208	60.047
A90	121.916	89.755
B0	31.899	0
B30	62.417	30.518
B45	77.964	46.065
B60	92.185	60.286
B90	121.632	89.733
Average of I-beams A and B		
0	32.030	
30	62.302	
45	77.845	
60	92.197	
90	121.774	

6.2.1 Binning Segmentation: Pose and Pose Errors of I-Beams

Tables 6.2, 6.3, and 6.4 give the results of the pose determination based on the binning segmentation strategy. Table 6.2 gives the letter of the I-beam identified and a summary of the computed errors between the SMS measured values of the four top flange points and the values estimated using the binning segmentation algorithm. It gives the mean errors and standard deviations of the four flange point errors for each scanned I-beam at each density, repetition, and floor angle.

At the 60^o angle for I-beam A, there is an error in identifying the size of the I-beam. This is seen by the large standard deviation of the *X* error in the medium scan density. Furthermore, at 90^o for I-beam A, all of the errors and standard deviation are too large to identify the size of the I-beam correctly. For I-beam B, at the medium point density at 60^o, although the I-beam is correctly identified, the identification is suspect due to the large variation in the *Y*-coordinate error. A similar conclusion can be raised about the 90^o values.

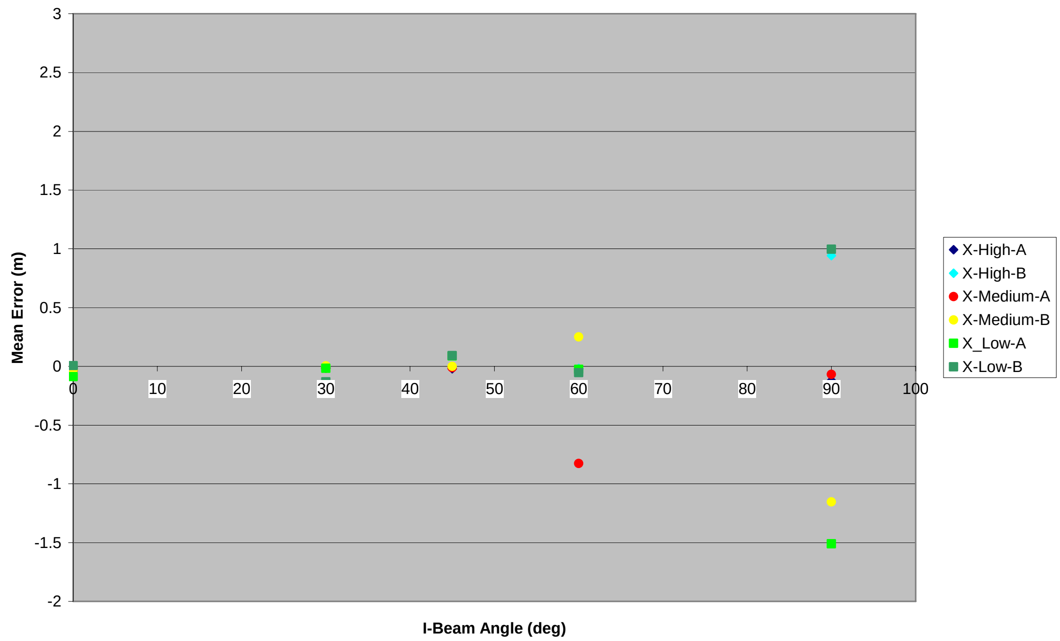
It appears that the binning segmentation process was not able to consistently determine the I-beam type for low point densities (20 gon) and when the longitudinal axis of the object was aligned with the scan direction, i.e., I-beam rotation of 90º. Identification errors also occurred for a few cases at the medium point density setting. A possible reason for this might have been the sizing of the voxels, but when smaller voxels were used, the I-beam was still incorrectly identified.

The mean errors and the standard deviations of the errors of the coordinates of the four corners of the top flange are plotted in Figures 6.1 and 6.2. In the legend in the figures, “X-High-A” refers to the errors of the *X*-coordinate, high density, I-Beam A – similarly for the remaining notation. The following observations are made:

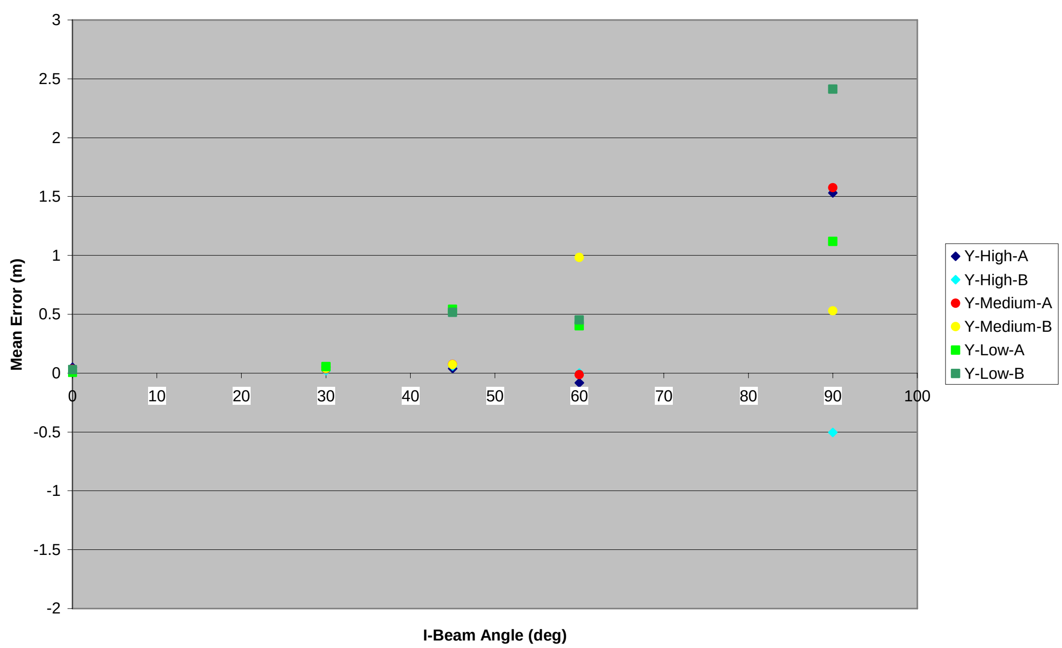
- *Z*-coordinate has the lowest error, followed by the *X*- and then the *Y*-coordinate
- The errors are stable and are generally within ± 5 cm for I-beam rotations of 0º and 30º and increase for angles greater than 30º.
- *Z*-coordinate has the lowest noise (standard deviation)
- For the error and standard deviation, no trends were observed as a function of the object – I-beam A or B.
- In general, for I-beam rotations greater than 30º, the errors for all coordinates (*x*, *y*, *z*) for the high density scans are smaller than that those for the medium and low densities.

Table 6.2. Binning Method: Summary of mean errors and standard deviations between the computed positions and measured positions at the four top flange corners.

File Prefix	Beam Identified as	Error Statistics at Top Flange Corners (m)						
		Point Density	**Mean X**	**Std. Dev. X**	**Mean Y**	**Std. Dev. Y**	**Mean Z**	**Std. Dev. Z**
A0	A	High	-0.0140	0.0164	0.0503	0.0131	-0.0155	0.0021
A0	A	Medium	-0.0332	0.0363	0.0343	0.0230	-0.0128	0.0046
A0	A	Low	-0.0884	0.0604	0.0056	0.0366	-0.0092	0.0035
A30	A	High	-0.0149	0.0151	0.0512	0.0214	-0.0146	0.0042
A30	A	Medium	-0.0157	0.0184	0.0353	0.0154	-0.0096	0.0051
A30	A/B	Low	-0.0173	0.1679	0.0545	0.2730	0.1242	0.1915
A45	A	High	-0.0194	0.0064	0.0383	0.0178	-0.0143	0.0027
A45	A	Medium	-0.0124	0.008	0.0739	0.0138	0.1305	0.2077
A45	B	Low	0.0883	0.1540	0.5425	0.6205	0.3828	0.0069
A60	A	High	-0.0198	0.0170	-0.0834	0.0515	-0.0140	0.0047
A60	A/B	Medium	-0.8254	1.2228	-0.0136	0.6482	-0.1026	0.4676
A60	B	Low	-0.0265	0.0741	0.4016	0.7262	0.3909	0.0090
A90	B	High	-0.1195	1.8327	1.5313	1.8304	-0.0444	1.0683
A90	B	Medium	-0.0681	1.9032	1.5746	1.6468	0.0103	1.0057
A90	B	Low	-1.5099	1.9399	1.1187	2.1873	-0.2969	1.1928
B0	B	High	-0.02314	0.0082	0.01732	0.0089	-0.018	0.0054
B0	B	Medium	-0.02845	0.0093	0.01529	0.0141	-0.0086	0.0078
B0	B	Low	0.006813	0.0163	0.0293	0.0131	-0.0027	0.0107
B30	B	High	-0.00012	0.0154	0.02036	0.0150	-0.0156	0.0095
B30	B	Medium	0.005069	0.02	0.03992	0.0215	-0.0112	0.0094
B30	B	Low	-0.1321	0.1875	-0.2172	0.3862	0.12996	0.1986
B45	B	High	-0.00109	0.0069	0.06193	0.0171	-0.0117	0.0071
B45	B	Medium	-0.00085	0.0133	0.07232	0.0310	-0.0065	0.0069
B45	B	Low	0.092634	0.0485	0.51586	0.1698	0.12031	0.1778
B60	B	High	-0.01769	0.0432	0.00146	0.0358	0.23773	0.1805
B60	B	Medium	0.249859	0.4234	0.98263	1.3417	0.11587	0.6701
B60	B	Low	-0.05444	0.0447	0.45195	0.0849	0.37554	0.0215
B90	B	High	0.942439	0.4376	-0.5033	2.4195	-0.1828	1.0574
B90	B	Medium	-1.15306	1.7427	0.5291	1.5149	-0.1644	0.9867
B90	B	Low	0.997166	0.6117	2.4117	0.9675	-0.0704	1.1262

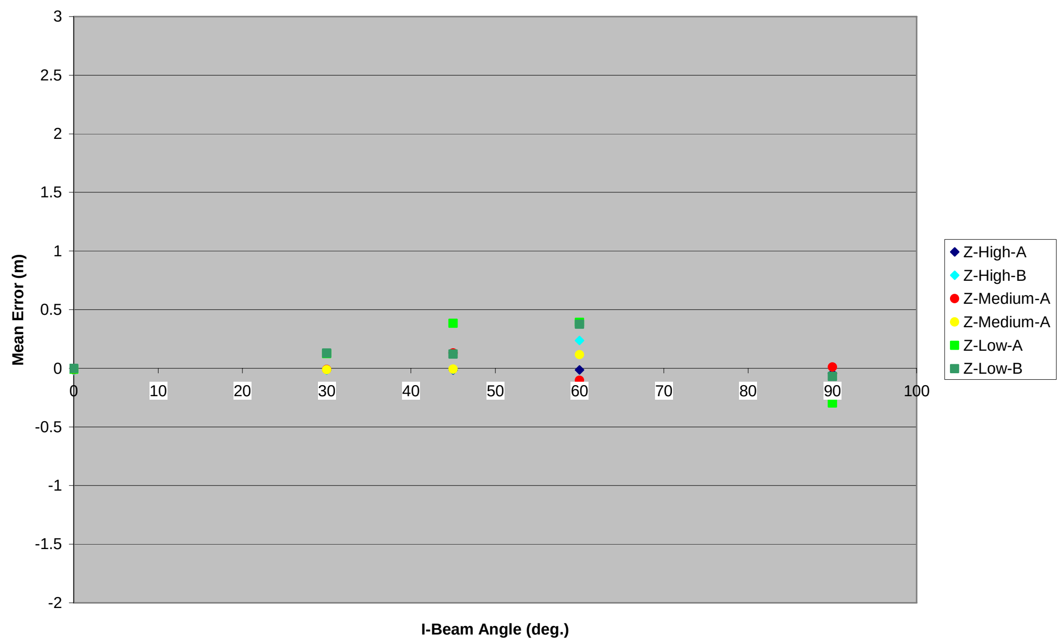


a. Mean Error in X-Coordinate.



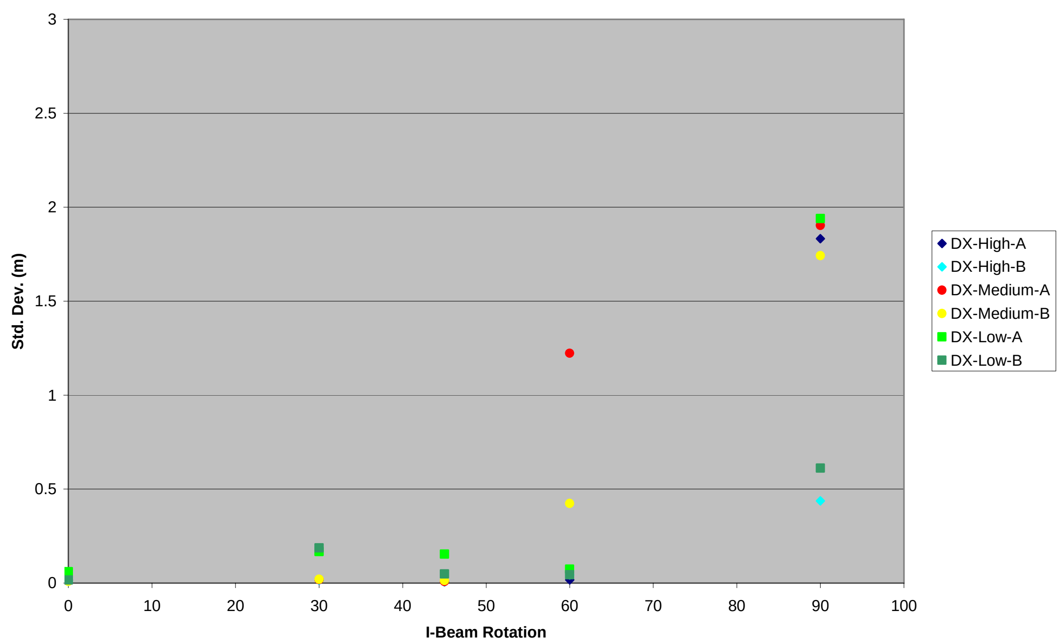
b. Mean Error in Y-Coordinate.

Figure 6.1. Binning Method: Mean Error in the Coordinates.



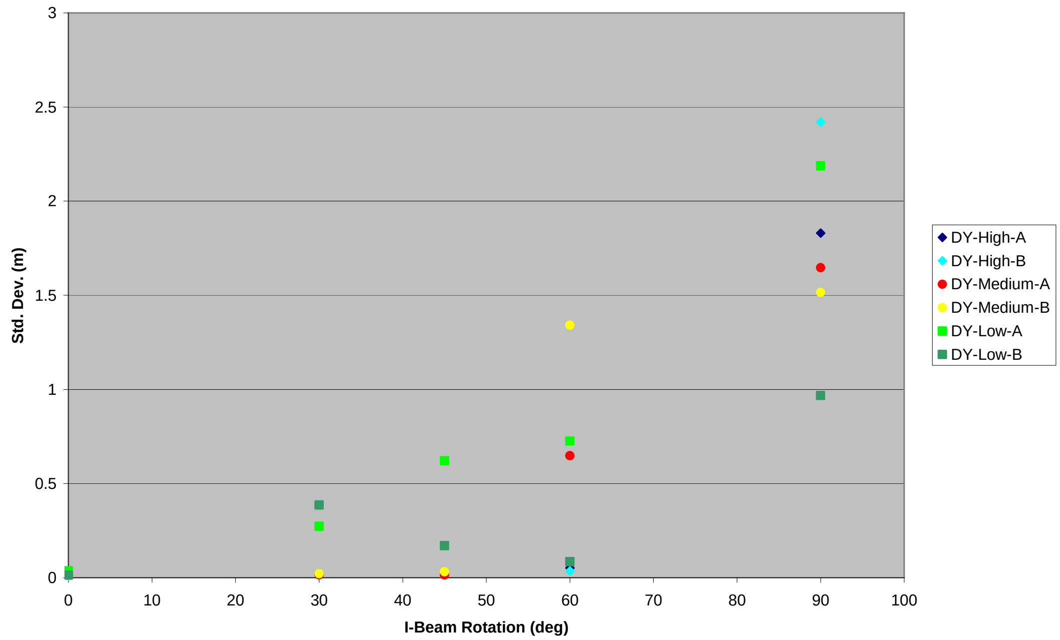
c. Mean Error in Z-Coordinate.

Figure 6.1. (cont.) Binning Method: Mean Errors of the Coordinates.

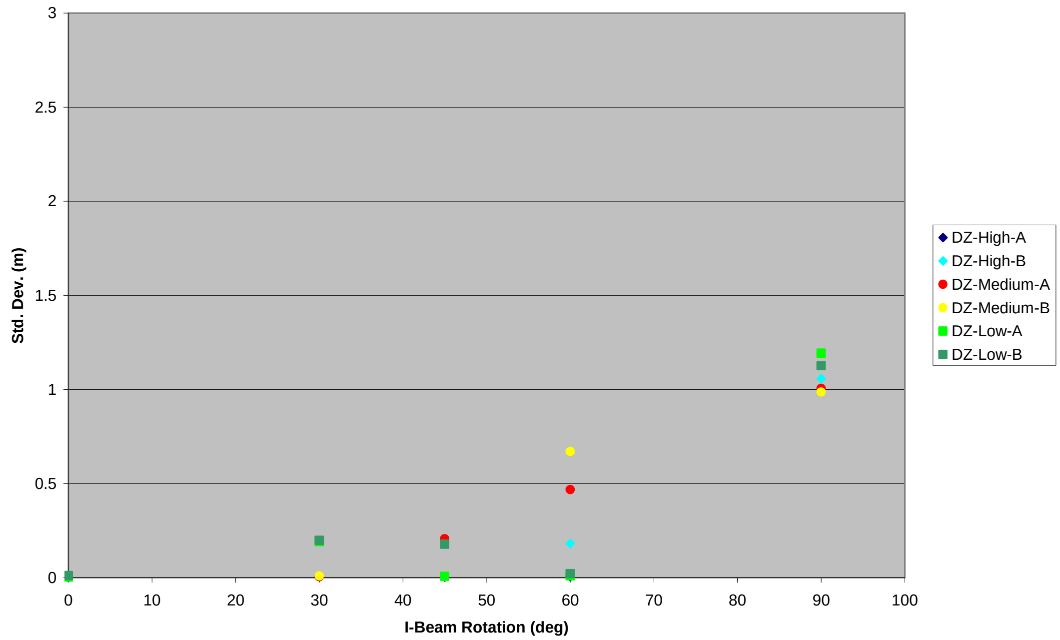


a. Standard Deviation of Errors in X-Coordinate.

Figure 6.2. Binning Method: Standard Deviation of Errors of the Coordinates.



b. Standard Deviation of Errors in Y-Coordinate.



c. Standard Deviation of Errors in Z-Coordinate.

Figure 6.2. (cont.) Binning Method: Standard Deviation of Errors of the Coordinates.

Table 6.3. Binning Method: Estimated orientation relative to the scanner's negative *X*-axis, the axis pointing directly in front of the scanner.

File Prefix	**Point Density**	**Calculated Angle (º)**		**Orientation Error = (Calc. – Ref. §) (º)**
		Mean	**Std. Dev.**	
A0	High	32.121	0.1209	-0.040
A0	Medium	32.103	0.3186	-0.058
A0	Low	32.124	0.2016	-0.037
A30	High	62.457	0.1275	0.271
A30	Medium	62.562	0.1317	0.376
A30	Low	62.606	1.2439	0.420
A45	High	77.851	0.1643	0.126
A45	Medium	77.558	0.2524	-0.167
A45	Low	78.425	1.0176	0.700
A60	High	91.701	0.4104	-0.507
A60	Medium	108.16	28.897	15.952
A60	Low	90	0	-2.208
A90	High	101.92	85.356	-19.996
A90	Medium	140.49	24.652	18.574
A90	Low	NA	NA	NA
B0	High	32.328	0.3451	0.429
B0	Medium	32.42	0.6283	0.521
B0	Low	32.108	0.6025	0.209
B30	High	63.203	0.1168	0.786
B30	Medium	63.327	0.2738	0.910
B30	Low	64.043	2.2729	1.626
B45	High	78.253	0.1907	0.289
B45	Medium	78.501	0.1438	0.537
B45	Low	79.167	0.8732	1.203
B60	High	94.372	0.4394	2.187
B60	Medium	90.09	5.7058	-2.095
B60	Low	90	0	-2.185
B90	High	108.87	73.112	-12.762
B90	Medium	154.86	17.804	32.228
B90	Low	NA	NA	NA

§ Reference values taken from Table 6.1 for I-beams A and B.

Table 6.3 gives the predicted angles relative to the scanner coordinates. For both I-beams, both the 60^o and the 90^o cases are suspect. In the cases marked "NA" the identification procedure could not determine and angle and produced a numerically undefined value.

Table 6.4 contains the same information as Table 6.3 but the information is grouped by I-beam rotations and the values for I-beams A and B are combined. From the table, it can be seen that as the longitudinal axis of the object becomes more parallel to the direction of the scan, there is more noise in the calculated object orientation.

Table 6.4. Binning Method: Comparison of the predicted orientation relative to the scanner as a function of I-beam Rotation angle and Point Density.

I-Beam Rotation (º)	Point Density	Calculated Orientation		Orientation Error = (Calc. – Ref. §)
		Mean (º)	Std. Dev. (º)	(º)
0	High	32.225	0.146	0.195
0	Medium	32.261	0.224	0.231
0	Low	32.116	0.012	0.086
30	High	62.830	0.527	0.528
30	Medium	62.944	0.541	0.642
30	Low	63.325	1.016	1.023
45	High	78.052	0.285	0.207
45	Medium	78.030	0.667	0.185
45	Low	78.796	0.525	0.951
60	High	93.037	1.888	0.840
60	Medium	99.125	12.778	6.928
60	Low	90.000	0.000	-2.197
90	High	105.394	4.918	-16.380
90	Medium	147.677	10.164	25.903
90	Low	NA	NA	NA

§ Reference values taken from Table 6.1 – Average of I-beams A and B.

Table 6.5 summarizes the location of the center-of-mass of the scanned I-beam data for the three scan densities, both I-beams, and the five I-beam rotations. Table 6.6 summarizes the center-of-mass locations by densities.

In Table 6.5, the standard deviation in the medium 60^o scan case for the *X*-coordinate is large which may be an indication of outliers in the data. This is consistent with the results from Table 6.2. Furthermore, the 90^o values are also suspect for the *X* and *Y* coordinates due to the large

Table 6.5. Binning Method: Summary of predicted center-of-mass of I-beams.

File Prefix	Point Density	Location of Center-of-Mass (m)					
		X	Std. Dev. X	Y	Std. Dev. Y	Z	Std. Dev. Z
A0	High	-3.6152	0.0169	5.7655	0.0118	-1.4025	0.0011
A0	Medium	-3.5956	0.0368	5.7812	0.0216	-1.4051	0.0005
A0	Low	-3.5423	0.0604	5.8120	0.0367	-1.4088	0.0010
A30	High	-3.6444	0.0133	5.7275	0.0211	-1.4045	0.0017
A30	Medium	-3.6419	0.0153	5.7423	0.0145	-1.4094	0.0018
A30	Low	-3.6424	0.076	5.7239	0.1393	-1.4128	0.0022
A45	High	-3.6371	0.0033	5.7182	0.0177	-1.4053	0.0009
A45	Medium	-3.6451	0.0027	5.6834	0.0138	-1.4104	0.0016
A45	Low	-3.7467	0.0932	5.2158	0.4120	-1.4111	0.0039
A60	High	-3.6462	0.0017	5.8110	0.0514	-1.4052	0.0004
A60	Medium	-2.8303	1.1941	5.6961	0.0831	-1.2854	0.1855
A60	Low	-3.6414	0.0098	5.3262	0.5469	-1.4188	0.0076
A90	High	-3.5153	1.7450	4.1024	1.2659	-1.1985	0.1657
A90	Medium	-3.5036	1.7273	4.0906	1.2579	-1.1838	0.1392
A90	Low	-3.5496	1.7628	4.4097	1.7046	-0.9124	0.4146
B0	High	-3.6218	0.0065	5.8187	0.0018	-1.4297	0.0006
B0	Medium	-3.6173	0.0051	5.8216	0.0063	-1.4392	0.0011
B0	Low	-3.6528	0.0153	5.8063	0.0091	-1.4450	0.0030
B30	High	-3.6823	0.0070	5.7760	0.0131	-1.4317	0.0019
B30	Medium	-3.6869	0.0148	5.7564	0.0206	-1.4360	0.0013
B30	Low	-3.5532	0.1809	6.0119	0.3834	-1.4561	0.0168
B45	High	-3.6932	0.0023	5.6943	0.0169	-1.4356	0.0012
B45	Medium	-3.6947	0.0083	5.6839	0.0313	-1.4408	0.0010
B45	Low	-3.7904	0.0400	5.2396	0.1709	-1.4464	0.0048
B60	High	-3.6741	0.0033	5.7199	0.0354	-1.4432	0.0017
B60	Medium	-3.9936	0.4971	4.7711	1.1425	-1.3811	0.1009
B60	Low	-3.6375	0.0140	5.2742	0.0839	-1.4599	0.0061
B90	High	-4.6523	0.0360	6.2020	2.1903	-1.0560	0.1836
B90	Medium	-2.4241	1.6568	5.0149	1.2996	-1.1562	0.1099
B90	Low	-4.6535	0.0024	3.2007	0.0534	-1.1954	0.1400

variability as indicated by the standard deviation. In the case of I-beam B, the medium 60^o scan is suspect in the *Y*-coordinate. Again, the 90^o cases are suspect in the *X* and *Y* coordinates.

Table 6.6. Binning Method: Variability of center-of-mass vs. point density.

Point Density	X (m)	Std. Dev. X (m)	Y (m)	Std. Dev. Y (m)	Z (m)	Std. Dev. Z (m)
High	-3.73819	0.3250	5.6336	0.5576	-1.3612	0.1287
Medium	-3.46333	0.4682	5.4041	0.5844	-1.3547	0.1075
Low	-3.74097	0.3308	5.2020	0.8305	-1.3567	0.1740

6.2.2 TIN Segmentation: Pose and Pose Errors

Tables 6.7 through 6.11 give the results of the pose determination based on the TIN segmentation method. These tables parallel Tables 6.2 through 6.6. Table 6.7 shows that segmentation by triangulation performs better at the 60^o cases but similar to the Binning method, the results for the 90^o cases are suspect. The data in Table 6.7 are shown graphically in Figures 6.3 and 6.4. The scales in these figures were chosen so that they were the same as those in Figures 6.1 and 6.2 so that a comparison between the two methods of segmentation could easily be made. Some observations about the pose and pose errors using the TIN methods are:

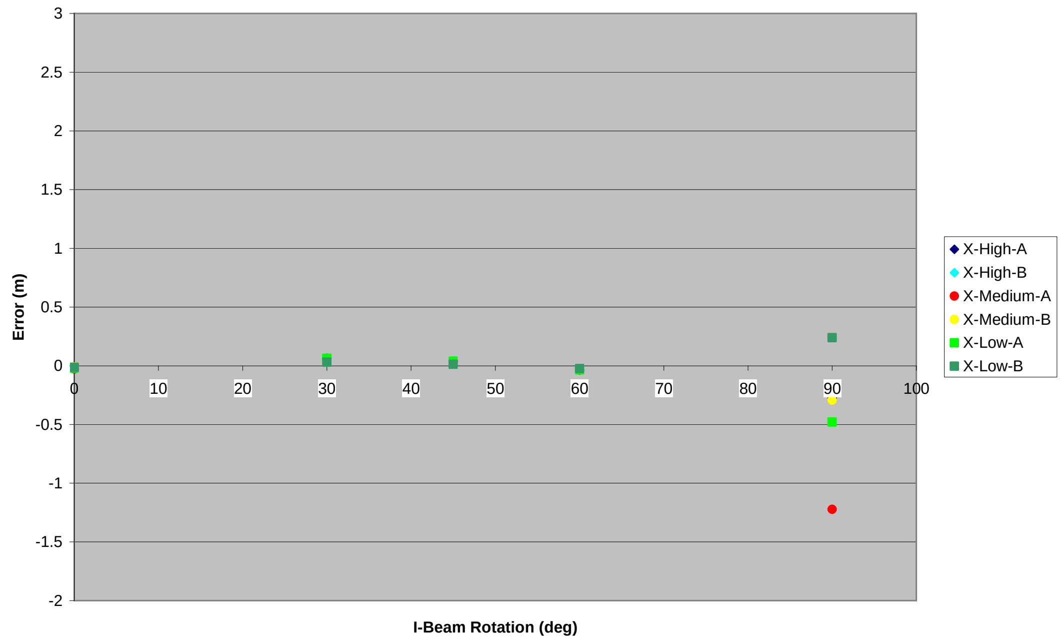
- Z-coordinate has the lowest error, followed by the X- and then the Y-coordinate – similar to the binning method
- For I-beam rotations from 0º to 60º, the Z-coordinate errors were within ±2 cm and X-coordinate errors were within ±4 cm (excluding the data for 30º).
- Z-coordinate has the lowest noise – similar to the binning method
- For the error and standard deviation, no trends were observed as a function of the object, i.e., I-beam A or B – similar to the binning method

Some observations based on the comparison of the errors and standard deviations between the binning and TIN methods the data:

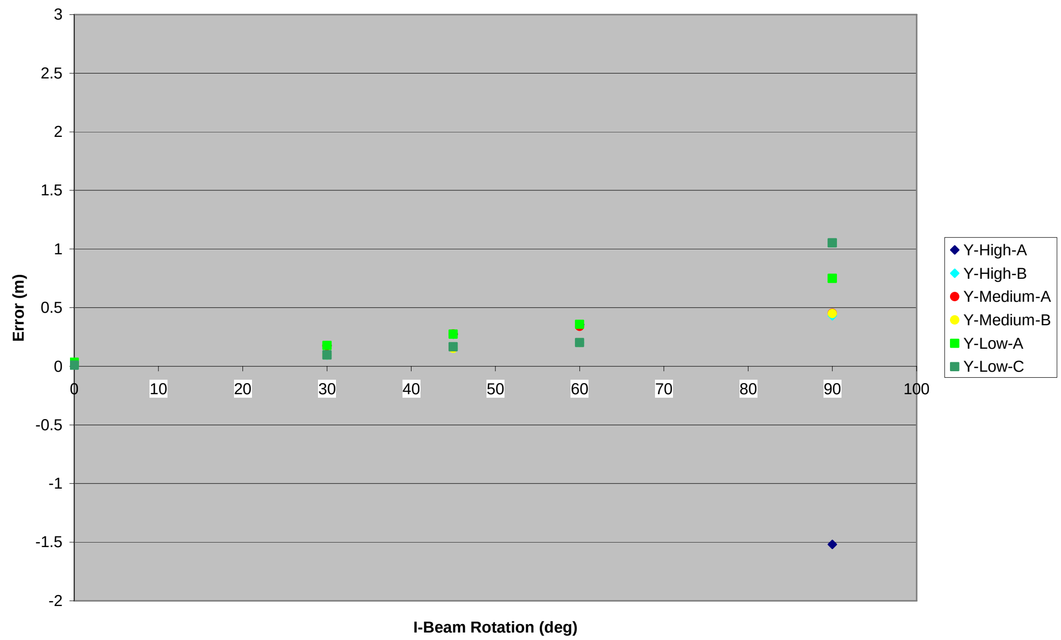
- The data coordinates are less noisy when using the TIN method – especially for I-beam rotations in the 30º to 60º range. (See Figure 6.5)
- The Z-coordinate errors are smaller for the TIN method vs. the bin method for I-beam rotations in the 30º to 60º range. No clear trends were observed for the X- and Y-coordinate errors.

Table 6.7. TIN Method: Summary of mean errors and standard deviations between the computed positions and measured positions at the four top flange corners.

File Prefix	I-Beam Identified as	Error Statistics at Top Flange Corners (m)						
		Point Density	Mean X	Std. Dev. X	Mean Y	Std. Dev. Y	Mean Z	Std. Dev. Z
A0	A	High	-0.0206	0.0031	0.0311	0.004	0.0004	0.1216
A0	A	Medium	-0.0283	0.0042	0.0285	0.0041	0.0006	0.0090
A0	A	Low	-0.0204	0.0059	0.0342	0.0059	0.0031	0.0088
A30	A	High	0.0647	0.0027	0.1771	0.0043	0.0008	0.0094
A30	A	Medium	0.0620	0.0027	0.1748	0.0027	0.0027	0.0078
A30	A	Low	0.0635	0.0064	0.1776	0.0080	0.0048	0.0085
A45	A	High	0.0368	0.1052	0.2733	0.0228	0.0017	0.0101
A45	A	Medium	0.0369	0.1052	0.2753	0.0229	0.0037	0.0105
A45	A	Low	0.0382	0.1053	0.2727	0.0258	0.0067	0.0091
A60	A	High	-0.0363	0.1859	0.3561	0.0079	0.0057	0.0083
A60	A	Medium	-0.0361	0.186	0.3401	0.0076	0.0077	0.0053
A60	A	Low	-0.0362	0.1859	0.3580	0.0082	0.0098	0.0071
A90	B	High	-0.2760	1.8893	-1.5199	1.9585	-0.1333	0.9077
A90	B	Medium	-1.2228	1.1232	0.4525	1.5728	0.2258	0.3661
A90	A/B	Low	-0.4795	0.2271	0.7491	0.3281	0.4269	0.0725
B0	B	High	-0.0125	0.1393	0.0136	0.2207	0.0074	0.0217
B0	B	Medium	-0.0105	0.1397	0.0170	0.2212	0.0079	0.0180
B0	B	Low	-0.0133	0.1399	0.0101	0.2209	0.0118	0.0210
B30	B	High	0.0309	0.0050	0.0943	0.0043	0.0071	0.0212
B30	B	Medium	0.0300	0.1349	0.0969	0.0697	0.0079	0.0172
B30	B	Low	0.0307	0.0030	0.0966	0.0070	0.0112	0.0195
B45	B	High	0.0139	0.0090	0.1520	0.0041	0.0088	0.0185
B45	B	Medium	0.0126	0.1488	0.1521	0.0315	0.0103	0.0144
B45	B	Low	0.0131	0.2106	0.1657	0.0448	0.0143	0.0107
B60	B	High	-0.0278	0.1839	0.2026	0.3909	-0.0044	0.0158
B60	B	Medium	-0.0310	0.2159	0.2020	0.0090	0.0138	0.0083
B60	B	Low	-0.0254	0.2157	0.2026	0.0144	0.0168	0.0136
B90	B	High	-0.2563	0.0194	0.4302	0.0105	0.3829	0.0317
B90	B	Medium	-0.2929	0.0185	0.4499	0.0121	0.3778	0.0422
B90	B	Low	0.2396	0.7957	1.0517	0.9920	0.2639	0.6388

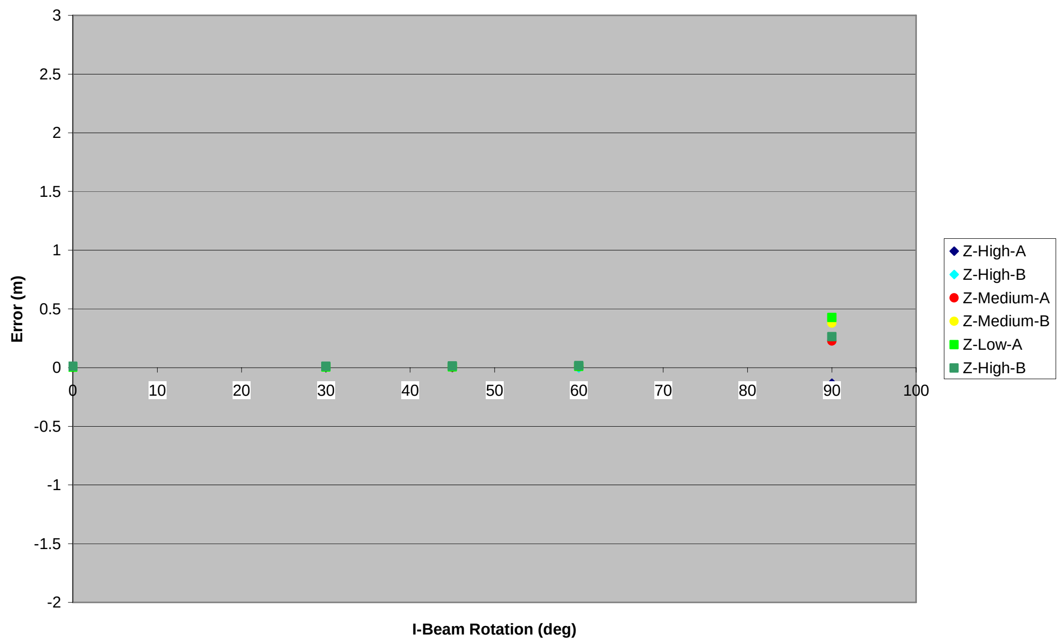


a. Mean Error in X-Coordinate.



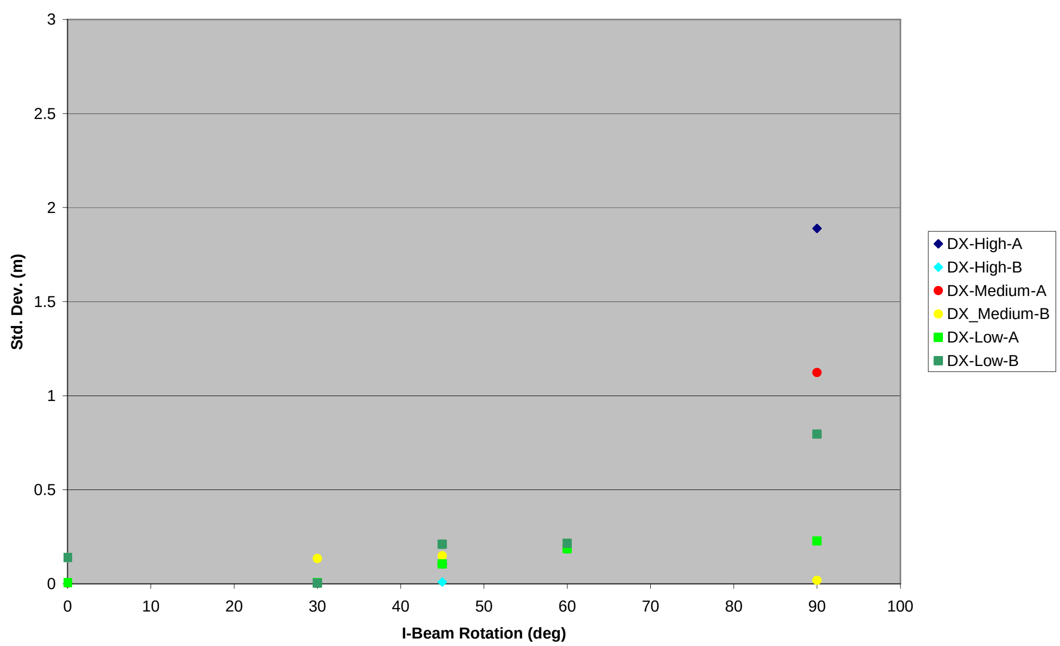
b. Mean Error in Y-Coordinate.

Figure 6.3. TIN Method: Mean Error in the Coordinates.



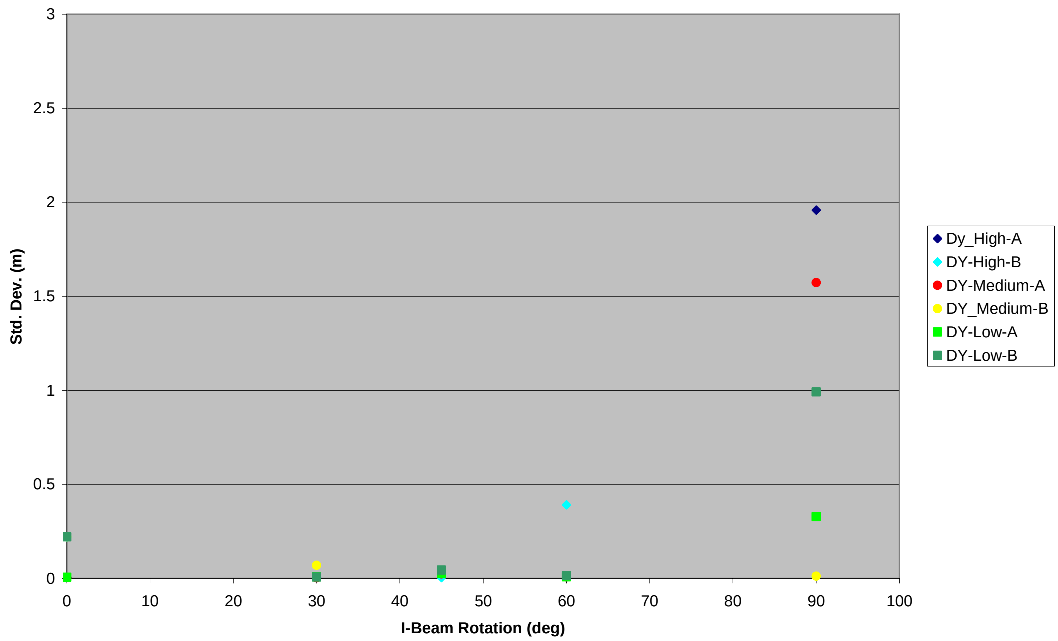
c. Mean Error in Z-Coordinate.

Figure 6.3. (cont.) TIN Method: Mean Error in the Coordinates.

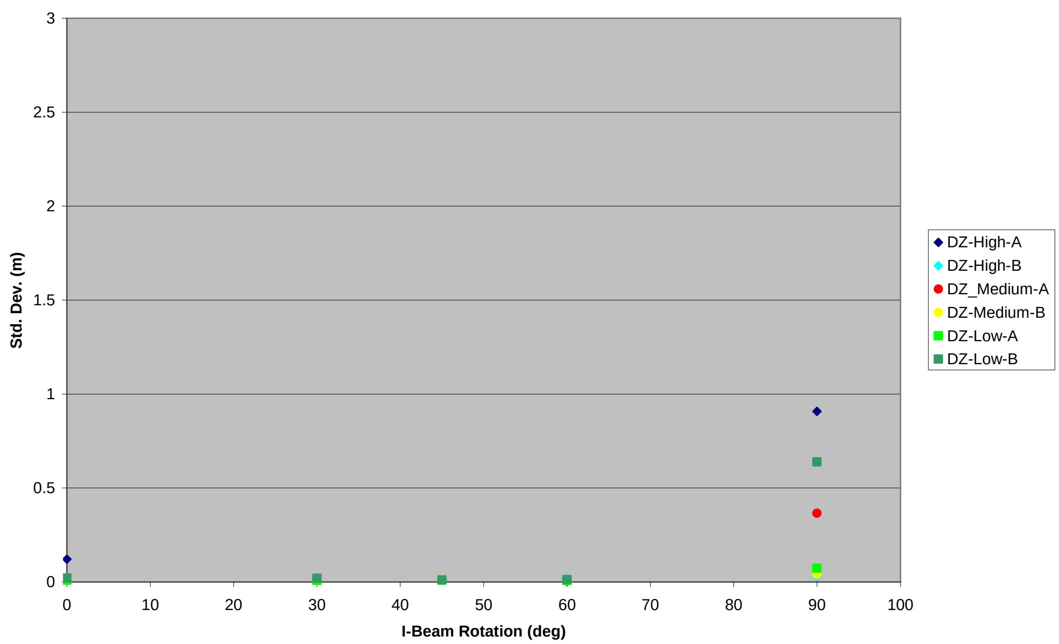


a. Standard Deviation of Error in X-Coordinate.

Figure 6.4. TIN Method: Standard Deviation of Errors of the Coordinates.

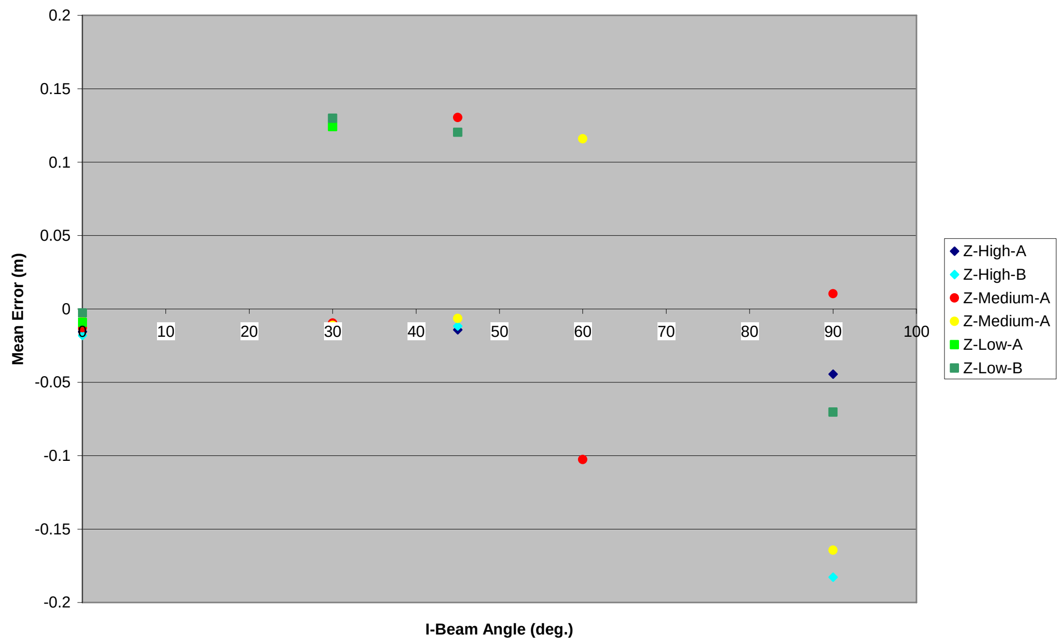


b. Standard Deviation of Error in Y-Coordinate.

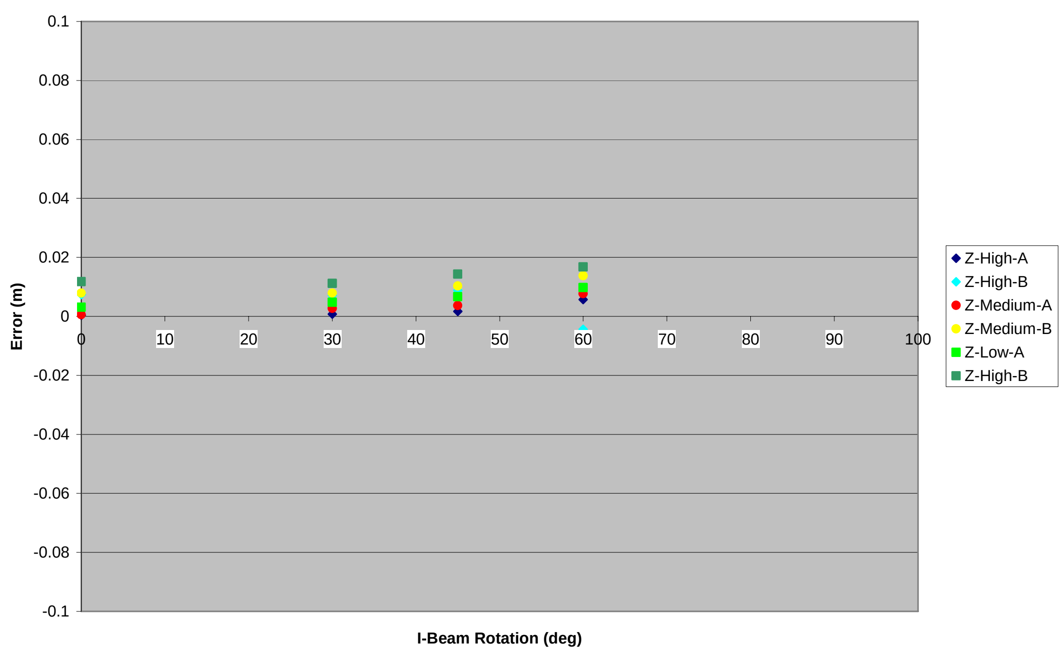


c. Standard Deviation of Error in Z-Coordinate.

Figure 6.4. (cont.) TIN Method: Standard Deviation of Errors of the Coordinates.



a. Bin Method: Mean Error in Z-Coordinate.



b. TIN Method: Mean Error in Z-Coordinate.

Figure 6.5. Comparison of Bin and TIN Methods: Mean Error in Z-Coordinate – enlarged scale.

Table 6.8. TIN Method: Estimated orientation relative to the scanner's negative *X*-axis, the axis pointing directly in front of the scanner.

File Prefix	Point Density	Calculated Angle (º)		Orientation Error = (Calc. – Ref. §) (º)
		Mean	Std. Dev.	
A0	High	32.271	0.022	0.110
A0	Medium	32.262	0.002	0.101
A0	Low	32.103	0.071	-0.058
A30	High	62.194	0.018	0.008
A30	Medium	62.221	0.020	0.035
A30	Low	62.288	0.050	0.102
A45	High	77.557	0.014	-0.168
A45	Medium	77.624	0.042	-0.101
A45	Low	77.596	0.114	-0.129
A60	High	91.989	0.006	-0.219
A60	Medium	91.904	0.029	-0.304
A60	Low	92.056	0.004	-0.152
A90	High	92.257	5.595	-29.659
A90	Medium	109.428	16.299	-12.488
A90	Low	120.644	0.705	-1.272
B0	High	32.213	0.104	0.314
B0	Medium	32.233	0.079	0.334
B0	Low	32.420	0.054	0.521
B30	High	62.267	0.145	-0.150
B30	Medium	62.300	0.182	-0.117
B30	Low	62.481	0.048	0.064
B45	High	77.461	0.093	-0.503
B45	Medium	77.612	0.087	-0.352
B45	Low	77.501	0.239	-0.463
B60	High	91.387	0.065	-0.798
B60	Medium	91.434	0.143	-0.751
B60	Low	91.433	0.182	-0.752
B90	High	121.516	0.865	-0.116
B90	Medium	120.634	0.226	-0.998
B90	Low	129.638	12.196	8.006

§ Reference values taken from Table 6.1 for I-beams A and B.

Table 6.8 parallels Table 6.3 by presenting the I-beam orientation and its variability. Again, the triangulation segmentation performs better in the 60° cases.

Table 6.9 contains the same information as Table 6.8 but the information is grouped by I-beam orientations. From the table, it can be seen that as the longitudinal axis of the object becomes more parallel to the direction of the scan, there is more noise in the calculated object orientation. This trend is also observed for the bin method.

Table 6.9. TIN Method: Comparison of the predicted orientation relative to the scanner as a function of I-beam rotation angle and point density.

I-Beam Rotation (°)	**Point Density**	**Calculated Angle (°)**		**Orientation Error = (Calc. – Ref. §) (°)**
		Mean	**Std. Dev.**	
0	High	32.242	0.041	0.212
0	Medium	32.247	0.020	0.217
0	Low	32.261	0.224	0.231
30	High	62.230	0.052	-0.072
30	Medium	62.260	0.056	-0.042
30	Low	62.385	0.136	0.083
45	High	77.509	0.068	-0.336
45	Medium	77.618	0.009	-0.227
45	Low	77.548	0.067	-0.297
60	High	91.688	0.426	-0.509
60	Medium	91.669	0.332	-0.528
60	Low	91.744	0.440	-0.453
90	High	106.886	20.689	-14.888
90	Medium	115.031	7.923	-6.743
90	Low	125.141	6.360	3.367

§ Reference values taken from Table 6.1 – Average of I-beams A and B.

A comparison of the orientation errors as a function of the I-beam rotation is shown in Figure 6.6. From this figure it can be seen that TIN method appears to result in smaller orientation errors for I-beam rotations of ≥ 30°.

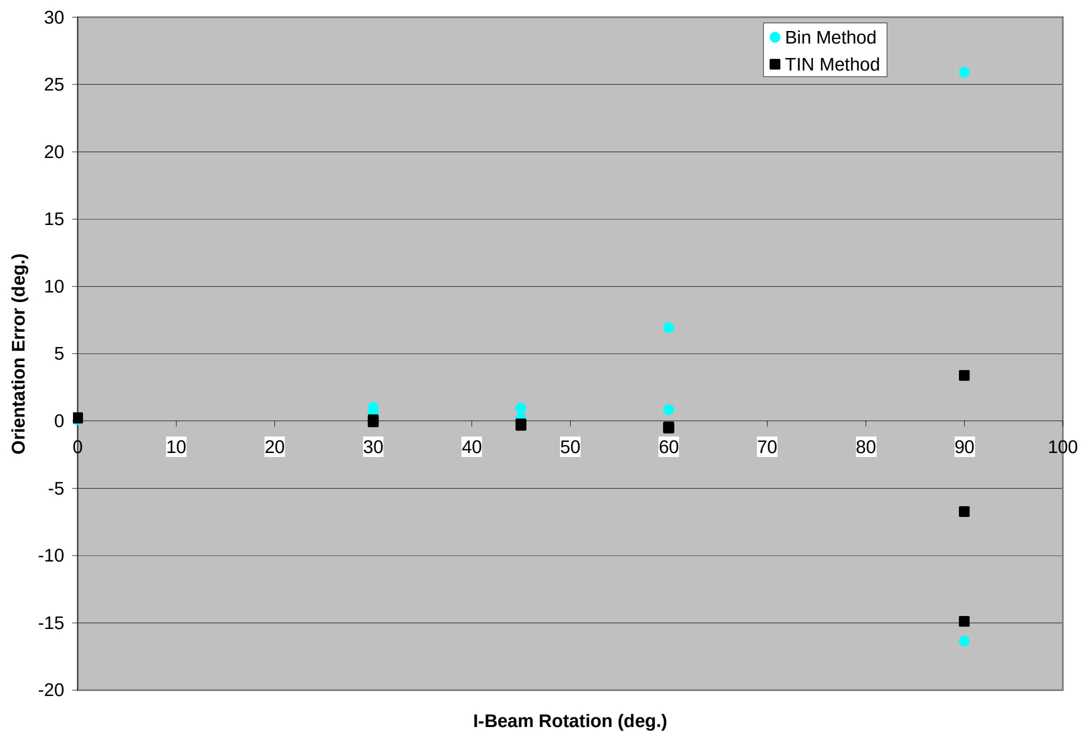


Figure 6.6. Orientation Error vs. I-beam Rotation.

Table 6.10 parallels Table 6.5 and presents the center-of-mass of the scanned data points using the TIN segmentation algorithm. This table shows that the TIN segmentation algorithm performs well in all cases except for the *X*- and *Y*-coordinates in the 90^o case for I-beam A in the low resolution scan.

Table 6.11 gives the variability of the center-of-mass as a function of the point density. There is no clear trend to show that the noise increases as the point density decreases.

Table 6.10. TIN Method: Predicted center-of-mass locations of I-beams.

File Prefix	Point Density	Location of Center-of-Mass (m)					
		X	Std. Dev. X	Y	Std. Dev. Y	Z	Std. Dev. Z
A0	High	-3.5953	0.0015	5.7665	0.0020	-1.4177	0.0003
A0	Medium	-3.5900	0.0034	5.7721	0.0022	-1.4181	0.0001
A0	Low	-3.5994	0.0063	5.7688	0.0048	-1.4206	0.0004
A30	High	-3.7079	0.0019	5.5889	0.0041	-1.419	0.0006
A30	Medium	-3.7070	0.0014	5.5920	0.0020	-1.421	0.0005
A30	Low	-3.7089	0.0029	5.5894	0.0084	-1.4232	0.0027

A45	High	-3.6791	0.0003	5.4739	0.0018	-1.4205	0.0003
A45	Medium	-3.6807	0.0004	5.4723	0.0036	-1.4226	0.0008
A45	Low	-3.6802	0.0016	5.4741	0.0125	-1.4255	0.0021
A60	High	-3.6170	0.0008	5.3646	0.0032	-1.4241	0.0010
A60	Medium	-3.6180	1.81E-05	5.3801	0.0027	-1.4262	0.0009
A60	Low	-3.6179	0.00058	5.3625	0.0036	-1.4282	0.0003
A90	High	-3.5087	1.7522	7.1705	0.7750	-1.1288	0.0441
A90	Medium	-2.4372	0.9615	5.2071	0.6278	-1.3681	0.1400
A90	Low	-3.1629	0.0247	4.9221	0.0454	-1.445	0.0074
B0	High	-3.6111	0.0009	5.7909	0.0006	-1.4521	0.0004
B0	Medium	-3.6159	0.0034	5.7917	0.0020	-1.4534	0.0007
B0	Low	-3.6111	0.0054	5.7967	0.0039	-1.4569	0.0013
B30	High	-3.6881	0.0024	5.6851	0.0041	-1.4524	0.0011
B30	Medium	-3.6898	0.0007	5.6839	0.0018	-1.4535	0.0013
B30	Low	-3.6900	0.0020	5.6838	0.0064	-1.4568	0.0009
B45	High	-3.6872	0.0007	5.5921	0.0023	-1.4544	0.0009
B45	Medium	-3.6890	0.0005	5.5934	0.0020	-1.4562	0.0006
B45	Low	-3.6929	0.0013	5.5806	0.0037	-1.4605	0.0031
B60	High	-3.6431	0.0014	5.5128	0.0004	-1.4607	0.0017
B60	Medium	-3.6444	0.00092	5.5132	0.0026	-1.4606	0.0008
B60	Low	-3.6437	0.0012	5.5129	0.0107	-1.4631	0.0013
B90	High	-3.3871	0.0098	5.2321	0.0086	-1.4664	0.0005
B90	Medium	-3.3653	0.0018	5.2063	0.0067	-1.4613	0.0024
B90	Low	-3.8440	0.7079	4.5701	0.9346	-1.4271	0.0559

Table 6.11. TIN Method: Variability of center-of-mass vs. point density.

Point Density	**X (m)**	**Std. Dev. X (m)**	**Y (m)**	**Std. Dev. Y (m)**	**Z (m)**	**Std. Dev. Z (m)**
High	-3.6125	0.0987	5.7177	0.5389	-1.4096	0.1004
Medium	-3.5037	0.3874	5.5212	0.2090	-1.4341	0.0292
Low	-3.6251	0.1769	5.4261	0.3905	-1.4407	0.0174

7. CONCLUSIONS

The objectives of this study were to segment objects from a point cloud obtained from a LADAR scan, identify the object from a set of known objects, and to develop an algorithm to determine the pose of that object. In this study, two approaches to segmenting data for potential object segmentation are presented: binning and TINs. The object identification involved the use of bounding boxes and a data set of potential objects in a scan region. The identification was made by comparing the bounding box of a potential object with that for the ideal object in the database. The measure-of-fit was the sum of squared errors of the vertices of the bounding boxes, and the object was associated with or "identified" as the ideal object whose bounding box yielded the smallest error. The method of principal axes was used to determine the pose of the object.

The parameters in this study were the point density (three point densities), the object size (two object sizes), and the rotation of the object relative to the scanner's scan direction (five object rotations). To quantify the uncertainty of the pose determination, reference points were obtained using a laser-based site measurement system (SMS). To obtain a common reference frame for the both the SMS and LADAR data, a transformation matrix had to be determined.

The results of this study have shown that classical methods in 3D image analysis can be used in a limited fashion to identify known simple objects. The difference between employing these classic methods in the context of LADAR scans involves first the order of magnitude of data acquired. LADARs can gather millions of points within a short period of time. Second, there is the problem of phantom points or mixed pixels. Third, there is a large amount of noise in the data. The large amount of data dictates the need to decide on a strategy of sub-sampling the data in order to make the computations tractable. The phantom pixels contribute to unwanted data values and require special filtering procedures. They also blur boundary edges in such a way that there is large uncertainty in identifying boundaries. This, of course, makes it difficult to use classic boundary detection algorithms. These algorithms are at the core of classic 3D image analysis.

This study has also shown that using principal components analysis on polygon models of segmented objects, when matched with polygon models of known objects, can be used to identify the pose of single isolated objects, such as I-beams. The success of the procedure depends mainly on the direction with which the object faces the scanner. The most successful identifications are obtained when the longitudinal axis of the object is perpendicular to the direction of scan of the LADAR. The closer the longitudinal axis of the object parallels the scan direction of the LADAR,

- the more difficult it is to determine the length of the object
- the noisier the data in terms of the x, y, z coordinates of the center-of-mass - similarly, the coordinates of the vertices of the top of the bounding box.

Based upon a search of the literature, to the authors' knowledge, this has been the first time that principle components analysis has been applied to determine the pose of objects in point clouds of LADAR scanned data.

With respect to point density, the results seem to indicate that, in general, the pose determination is more accurate for higher point densities. There were only 3 instances, about 3 % of all cases, when the I-beam was incorrectly identified, and each of these occurred for the lower point density data sets.

In the process to determine of the transformation matrix, it was found that there is large variability in determining the center of the reference spheres using the LADAR scanner. There is difficulty in fitting a sphere model through a set of points that are only located on one side of a sphere. There is further need to quantify how the uncertainty in identifying the center of the reference spheres propagates into the uncertainty of the object pose relative to the scanner.

REFERENCES

[1] Amann, M., Bosch, T., Lescure, M., Myllylä, R., and Rioux, M., [2001], "Laser ranging: a critical review of usual techniques for distance measurement," *Opt. Eng.*, 40 (1), 10 – 19.

[2] Bajaj, C. L., Bernardini, F., and Xu, G., [1995], 'Automatic Reconstruction of Surfaces and Scalar Fields from 3D Scans," *Computer Graphics Proceedings Annual Conference Series, SIGGRAPH 95,* 109-118.

[3] Barhak, J. and Fischer, A., [2001], "Parameterization and Reconstruction from 3D Scattered Points Based on Neural Netwwork and PDE Techniques," *IEEE Transactions on Visualization and Computer Graphics*, 7 (1), 1 – 16.

[4] Beraldin, J. A., Blais, F., Rioux, M., Cournoyer, L., Laurin, D. and MacLean, S. G., [1997], "Short and medium range 3-D sensing for space applications," *SPIE Aerosense '97, Proc. Visual Information Processsing VI,* 3074, 29 – 46.

[5] Beraldin, J. A., Blais, F., Rioux, M., Cournoyer, L., Laurin, D. and MacLean, S. G.,[2000], "Eye-safe digital 3-D sensing for space applications", *Opt. Eng.*, 39 (1), 196 – 211.

[6] Beraldin, J. A., Blais, F., Boulangere, P., Cournoyer, L., Domey, J., El-Hakim, S. F., Godin, G., Rioux, M., and Taylor, J., [2000], "Real world modelling through high resolution digital 3D imaging of objects and structures," *ISPRS Journal of Photogrammetry & Remote Sensing,* 55, 230 – 250.

[7] Bergevin, R., Laurendeau, D., and Poussart, D., [1995], "Registering Range Views of Multipart Objects," *Computer Vision and Image Processing*, 61 (1), 1 – 16.

[8] Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., and Taubin, G., "The Ball-Pivoting Algorithm for Surface Reconstruction," *IEEE Transactions on Visualization and Computer Graphics*, 5 (4), 349 – 359.

[9] Bernardini, F. and Rushmeier, H., [2000], "Strategies for registering range images from unknown camera positions," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE* , 3958, 200 – 206.

[10] Besl, P. J. and McKay, N. D., [1992], "A Method for Registration of 3_D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14, (2), 239 – 256.

[11] Bors, A. G. and Pitas, I, [1999], "Object Classification in 3-D Images Using Apha-Trimmed Mean Radial Basis Function Network," *IEEE Transactions on Image Processing*, 8 (12), 1744 – 1756.

[12] Boughorbal, F., Page, D. L., and Abidi, M. A., [2000], "Automatic reconstruction of large 3D models of real environments from unregistered data-sets," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE* , 3958, 234 – 241.

[13] Caelli, T., Osman, E., and West, G., [1998], "3D Shape Matching and Inspection Using Geometric Features and Relational Learning," *Computer Vision and Image Understanding*, 72, (3), 340 – 350.

[14] Chatzis, V. and Pitas, I., [2000], "A Generalized Fuzzy Mathematical Morphology and its Application in Robust 2-D and 3-D Object Representation," *IEEE Transactions on Image Processing*, 9 (10), 1798 – 1810.

[15] Cignoni, P., Montani, C., Puppo, E., Scopigno, R., [1997], "Multiresolution Representation and Visualization of Volume Data," *IEEE Transactions on Visualization and Computer Graphics*, 3 (4), 352 – 369.

[16] Cooper, M., Srivastava, A., and Miller, M., [1999], "Quadratic Analysis of Information Measures for Object Recognition," *Proc. SPIE*, 3721.

[17] Cunnington and Stoddart, A. J., [1999], "N-View Point Set Registration: A Comparison," *Proc. British Machine Vision Conference*, Nottingham, UK.

[18] Curless, B., [1999], "From Range Scans to 3D Models," Computer Graphics, 33 (4).

[19] Curless, B. and Levoy, M., [1996], "A Volumetric Method for Building Complex Models from Range Images,"*Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH 96,* 303 – 312.

[20] Dorai, C. and Jain, A. K., [1997], "COSMOS – A Representation Scheme for 3D Free-Form Objects," *IEEE Transactions on Patteren Analysis and Machine Intelligence*, 19 (10), 1115 – 1130.

[21] El-Hakim, S., [2000], "A Practical Approach to Creating Precise and Detailed 3D Models from Single and Multiple Views," *IAPRS, Vol. XXXIII,* Amsterdam, Book 5A, 202-209.

[22] Enders, R. H. and Shapiro, J. H., [1988], "Laser radar tracking theory", *SPIE Laser Radar II*I, 999, 192—215.

[23] Fan, T., Medioni, G., and Nevatia, R., [1989], "Recognizing 3-D Objects Using Surface Descriptions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, (11), 1140 – 1157.

[24] FIATECH, [2004], "Capital Projects Technology Roadmap: Element 4 Tactical Plan, Intelligent and Automated Construction Job Site (IACJS)".

[25] Gilsinn, D. E., Cheok, G. S., and Lytle, A. M, [2004], "Pose of I-beams for Construction Site Automation," *Proceedings 21st International Symposium on Automation and Robotics in Construction*, September 21-25, Jeju, Korea.

[26] Gilsinn, D. E., Cheok, G. S., and O'Leary, D. [2004], "Reconstructing Images of Bar Codes for Construction Site Object Recognition," *Automation in Construction*, 13 (1) January.

[27] Golnabi, H. [2000], "Design and operation of a laser scanning system," *Optics & Laser Technology*, 32, 267 – 272.

[28] Golub, G. H. and Van Loan, C. F.[1993], *Matrix Computations*, Johns Hopkins Universsity Press, 4th ed.

[29] Green, T. J., Jr., and Shapiro, J. H., [1994], "Detecting objects in three-dimensional laser radar range images," *Optical Engineering*, 33 (3), 865 – 874.

[30] Greenberg, S., Rotman, S. R., Guterman, H., and Gens, A., [2005], "Region-of-interest-based algorithm for automatic target detection in infrared images," *Optical Engineering*, 44 (7).

[31] Greenspan, M. A., [2002], "Geometric Probing of Dense Range Data", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, (4), 495 – 508.

[32] Haralick, R. M., Joo, H., Lee, C., Zhuang, X., Vaidya, V. G. and Kim, M. B., [1989], "Pose Estimation from Corresponding Point Data," *IEEE Transactions on Systems, Man, and Cybernetics*, 19, (6), 1426 – 1446.

[33] Hashemi, K. S., Hurst, P. T., and Oliver, J. N., [1994], "Sources of error in a laser range finder", *Rev. Sci. Instrum.*, 65 (10), 3165 -- 3171

[34] Hebert, M. and Krotkov, E., [1991], "3-D Measurements from Imaging Laser Radars: How Good are They?", *IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS '91*, Nov. 3-5, Osaka, Japan.

[35] Hemayed, E. E., Mostafa, M. G., and Farag, A. A., [2000], " 3D Object Reconstruction from a Sequence of Images Using Voxel Coloring," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE* , 3958, 207 – 214.

[36] Herbin, M. Venot, A., Devaux, J. Y., Walter, E., Lebruchec, J. F., Dubertret, L., and Roucayrol, J. C., [1989], "Automated Registration of Dissimilar Images: Application to Medical Imagery," *Computer Vision, Graphics, and Image Processing*, 47, 77 – 88.

[37] Hirsch, B. E., Thoben, K. D., Sheng, X., and Joppe, M., [1997], "Reconstruction of Volume Boundary from Layers by Triangulation," *Journal of Manufacturing Systems*, 16 (4), 249 – 259.

[38] Hoover, A., Jean-Baptiste, G. Jiang, X., Flynn, P. J., Bunke, H., Goldgof, D. B., Bowyer, K., Eggert, D. W., Fitzgibbon, A., and Fisher, R. B., [1996], "An Experimental Comparison of Range Image Segmentation Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18 (7), 673 – 689.

[39] Hoppe, H., DeRose, T., Duchamp, T., Halstead, M., Jin, H., McDonald, J., Schweitzer, J., and Tuetzle, W., "Piecewise Smooth Surface Reconstruction," *Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH 94*, 295 – 302.

[40] Hsieh, J., Liao, H. M., Fan, K., Ko, M., and Hung, Y., [1997], "Image Registration Using a New Edge-Based Approach," *Computer Vision and Image Understanding*, 67 (2), 112 – 130.

[41] Huber, D., Carmichael, O., Hebert, M., [2000], "3-D Map Reconstruction from Range Data," *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, 891 – 897.

[42] Hutchinson, B. A., Galbraith, R. L., Stann, B. L. Der, S. Z., [2003], "Simulation-based analysis of range and cross-range resolution requirements for the identification of vehicles in ladar imagery," *Opt. Eng*., 42 (9) 2734 – 2745.

[43] Huttenlocher, D. P. and Ullman, S., [1990], "Recognizing Solid Objects by Alignment with an Image," *International Journal of Computer Vision*, 5, (2), 195 – 212.

[44] Jiang, X. and Bunke, H., [1999], "Edge Detection in Range Images Based on Scan Line Approximation," *Computer Vision and Image Understanding*, 73 (2), 183 – 199.

[45] Johnson, A. E., [1997], "Spin-Images: A Representation for 3-D Surface Matching," Ph. D. Thesis, Carnegie Mellon University, Pittsburgh, PA, 15213.

[46] Johnson, A. E., Carmichel, O., Huber, D., and Hebert, M., [1998], "Toward a General 3-D Matching Engine: Multiple Models, Complex Scenes, and Efficient Data Filtering, " *Proc. Of the 1998 Image Understanding Workshop (IUW)*, 1097 – 1107.

[47] Johnson, A. E. and Hebert, M. [1999], "Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21 (5), 433 – 449.

[48] Juarez, E. L., Dumont, C., and Abidi, M. A., [2000], "Object Modeling in Multiple-Object 3D Scene Using Deformable Simplex Meshes," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE,* 3958, 144 – 152.

[49] Kanamaru, T., Yamada, K., Ichikawa, T., Naemura, T., Aizawa, K., and Saito, T., [200], "Acquisition of 3D Image Representation in Multimedia Ambiance Communication

Using 3D Laser Scanner and Digital Camera," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE* , 3958, 80 – 89.

[50] Kim, W. and Kak, A. C., [1991], "3-D Object Recognition Using Bipartite Matching Embedded in Discrete Relaxation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13, (3), 224 – 251.

[51] Krishnamurthy, V. and Levoy, M., [1996], "Fitting Smooth Surfaces to Dense Polygon Meshes," *SIGGRAPH 96*

[52] Krishnapuram, R. and Cassasent, D., [1989], "Determination of Three-Dimensional Object Location and Orientation from Range Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, (11), 1158 – 1167.

[53] Laboureux, X. and Häusler, G. [2001], "Localization and registration of three-dimensional objects in space – where are the limits?" *Applied Optics*, 49 (29), 5206 – 5216.

[54] Lawrence, J. and Witzgall, C., [2005] "A trace-Maximization Problem associated with the Registration of Spatial Points Sets," Preliminary draft.

[55] Lawson, C. L., [1977], "Software for C^1 surface interpolation," Mathematical Software III, J. R. Rice (Ed.), Academic Press, New York, 161 – 194.

[56] Lengyel, E., [2002], *Mathematics for 3D Game Programming and Computer Graphics*, Charles River Media, Inc., Hingham, Massachusetts.

[57] Leonardis, A., Jaklič, A., and Solina, F., [1997], "Superquadrics for Segmenting and Modeling Range Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19 (11), 1289 – 1295.

[58] Levy, A. and Lindenbaum, M., [2000], "Sequential Karhunen – Loeve Basis Extraction and its Application to Images," *IEEE Transactions on Image Processing*, 9 (8), 1371 – 1374.

[59] Liang, P. and Todhunter, J. S., [1990], "Representation and Recognition of Surface Shapes in Range Images: A Differential Geometry Approach," *Computer Vision, Graphics, and Image Processing*, 52, 78 – 109.

[60] Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L. F., Faust, N., and Turner, G. A., [1996], *Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH 96.*

[61] Lu, T. and Yun, D. Y. Y.,[2000], "Optimizing Triangular Mesh Generation from Range Images," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE,* 3958, 161 – 169.

[62] Lucchese, L., Doretto, G., and Cortelazzo, G. M., [2002], "A Frequency Domain Technique for Range Data Registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (11), 1468 – 1484.

[63] Luebke, D. and Erikson, C., [1997], "View-Dependent Simplification of Arbitrary Polygonal Environments," *Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH 97.*

[64] Lytle, A. M. and Saidi, K. S., [2004], "Proceedings of the Automated Steel Construction Workshop, June 6-7, 2002", *NISTIR 7108*, National Institute of Standards and Technology, Gaithersburg, MD.

[65] Marshall, D., Lukacs, G., and Martin, R., [2001], "Robust Segmentation of Primitives from Range Data in the Presence of Geometric Degeneracy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23 (3), 304 – 314.

[66] Marshall, S. J., Whiteford, D. N., and Rixon, R. C., [2001], "Assessing the Performance of 3D Whole Body Imaging Systems," *Proc. 6th Numéisation 3D/Scanning 2001 Congress.*

[67] Matoba, O. Tajahuerce, E., and Javidi, B., [2001], "Real-time three-dimensional object recognition with multiple perspectives imaging," *Applied Optics*, 40 (20), 3318 – 3325.

[68] Mirmehdi, M., Palmer, P. L., Kittler, J., and Dabis, H.,[1999], "Feedback Control Strategies for Object Recognition," *IEEE Transactions on Image Processing*, 8 (8), 1084 – 1101.

[69] Monga, O. and Benayoun, S., [1995], "Using Partial Derivatives of 3D Images to Extract Typical Surface Features," *Computer Vision and Image Processing*, 61 (2), 171 – 189.

[70] Montgomery, D. C. and Peck, E. A., [1992], *Introduction to Linear Regression Analysis*, John Wiley &Sons, Inc., New York.

[71] Nikolaidis, N. and Pitas, I., [2001], *3-D Image Processing Algorithms,* John Wiley & Sons, Inc. New York.

[72] Pollefeys, M., Koch, R., Vergauwen, M., Deknuydt, B., and Van Gool, L., [2000], "Three-dimensional scene reconstruction from images," *Three-Dimensional Image Capture and Applications III, Proceedings of SPIE,* 3958, 215 – 226.

[73] O'Leary, D. P., Oral Communication.

[74] Saidi, K.S., Lytle, A.M., Scott, N.A. and Stone, W.C.,[2005], "Developments Toward Automated Construction," *NISTIR 7264*, National Institute of Standards and Technology, Gaithersburg, MD.

[75] Surlier, D. and Bucher, A., [1995], "Surface-independent, theoretically exact bestfit for arbitrary sculptured, complex, or standard geometries," *Precision Engineering*, 17, 101 – 113.

[76] Stone, W. C., Juberts, M., Dagalakis, N, Stone, J, and Gorman, J., [2004], "Performance Analysis of Next-Generation LADAR for Manufacturing, Construction, and Mobility," *NISTIR 7117*, National Institute of Standards and Technology, Gaithersburg, MD, May.

[77] Soucy, G. and Ferrie, F. P., [1997], "Surface Recovery from Range Images Using Curvature and Motion Consistency," *Computer Vision and Image Understanding*, 65 (1), 1 – 18.

[78] Turk, G. and Levoy, M., [1994], "Zippered Polgon Meshes from Range Images," Computer Graphics Proceedings, Annual Conference Series, *SIGGRAPH 94*, 311 – 318.

[79] Verly, J. G., Delanoy, R. L., and Dudgeon, D. E., [1992], "Model-based system for automatic target recognition from forward-looking laser-radar imagery," *Optical Engineering*, 31 (12), 2540 – 2552.

[80] Weiss, I. and Ray, M., [2001], "Model-Based Recognition of 3D Objects from Single Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23 (2), 116 – 128.

[81] Welch, W. and Witkin, A., [1994], "Free-Form Shape Design Using Triangulated Surfaces," *Computer Graphics Proceedings, Annual Conference Series, SIGGRAPH 94*, 247 – 256.

[82] Whitaker, R. T. and Gregor, J., [2002], "A Maximum-Likelihood Surface Estimator for Dense Range Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (10), 1372 – 1387.

[83] Wheeler, M. D. and Ikeuchi, K., [1995], "Sensor Modeling, Probabilistic Hypothesis Generation, and Robust Localization for Object Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17 (3), 252 – 265.

[84] Williams, J. and Bennamoun, M., [2001], "Simultaneous Registration of Multiple Corresponding Point Sets," *Computer Vision and Image Understanding*, 81, 117 – 142.

[85] Witzgall, C., Bernal, J., and Cheok, G. S. [2001], "Registering 3D Point Clouds: An Experimental Evaluation," *NISTIR 6743*, National Institute of Standards and Technology, Gaithersburg, MD, May.

[86] Witzgall, C., Bernal, J., and Cheok, G. S. [2004], "TIN Techniques for Data Analysis and Surface Construction," *NISTIR 7078*, National Institute of Standards and Technology, Gaithersburg, MD, January.

[87] Witzgall, C., Bernal, J., and Cheok, G. S. [2004], "Triangular Meshing and Volume Determination," *NISTIR 7142*, National Institute of Standards and Technology, Gaithersburg, MD, July.

[88] Wurster, J., Stark, H., Olsen, E. T., and Kogler, K., [1995], "Remote object recognition by analysis of surface structure," *J. Opt. Soc. Am. A*, 12 (6), 1242 – 1253.

[89] Zhang, D. and Hebert, M., [1999], "Harmonic Maps and Their Applications in Surface Matching," *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '99*).

[90] Zhao, C. and Mohr, R., [1996], " Global Three-Dimensional Surface Reconstruction from Occluding Contours," *Computer Vision and Image Understanding*, 64 (1), 62 – 96.

[91] Zheng, Q., Der, S. Z., and Mahmoud, H. I., [2001], "Model-Based Target Recognition in Pulsed Ladar Imagery," *IEEE Transactions on Image Processing*, 10 (4), 565 – 572.

APPENDICES

A.1 PROGRAM LISTINGS

A.1.1 Object Identification by Binning Scripts

The first script is the main driver script for the entire binning segmentation of objects. The main script uses functions bin3d.m, princ_axes.m, get_vertical_string.m, Segment_objects.m, Object_test.m. The function Segment_objects calls functions push.m and pop.m for stack manipulation.

A.1.1.1 MAIN BINNING SEGMENTATION SCRIPT

```
% Begin segmentation algorithm by binning
clear all;
%*************************************************************
%
% A Pose Algorithm for I-beams
%
% Although this script is constructed to identify and isolated
% I-beam in a point cloud of data obtained by a LADAR scan,
% the principles used are quite general.
%
% 1. Automatic object recognition from a set of unstructured
%    data points requires the provision of some prior data
%    of what potential objects could possibly be in the
%    scene.
% 2. The massive amont of raw data provided by the LADAR
%    usually does not provide determination of edges of
%    objects due to the uncertainty of mixed pixels
% 3. The raw data needs to be binned into appropriately
%    selected 3D cells or voxels.
% 4. Once the data is binned into 3D voxels then it is
%    possible to eliminate voxels that contain mixed
%    pixels and other outlying data, such as ground hits.
% 5. Once these voxels have been eliminated by setting
%    their bin count to zero, the remaining voxels with
%    non-zero bin counts can be grouped into objects by
%    joining neighboring voxels with non-zero counts.
% 6. The objects can then be tested for having 3
%    dimensions. The assumption here is that a legitimate
%    I-beam has 3 definite dimensions. If an object fails
%    the 3 dimension test it's voxels are also eliminated.
%    The test is based on whether 4 points lay on a plane
%    or not.
% 7. Once legitimate 3D objects are determined then their
%    princpal axes can be determined. These identify the
%    longest, next longest, etc. axes. The directions
```

```
%    of these axes specify the pose of the object relative
%    to the scanner.
% 8. Once principal axes are determined for an object then
%    a bounding box can be placed around it that encloses all
%    of its voxel centers.
% 9. Data from a database of I-beams can be read and corners
%    determined so that bounding boxes can be formed around
%    the ideal beams.
% 10. These ideal bounding boxes can then be compared to the
%     bounding boxes of the scanned objects to determine
%     which I-beams have been scanned by size comparison
%     of bounding boxes.
%
%
%************************************************************
%
%Open file for Excel comma separated output
fid3 = fopen('All_scanfile_voxel_output_1.txt','w');
% set up header lines for the Excel file
fprintf(fid3,'File, I-Beam, 8 Point, Location, of, CM, Pose from, Site,, Metrology, System, ,Bounding, Box, , ,Error\n');
fprintf(fid3,'Name, Selected, Error, x,y,z, neg x axis (deg),Point,x,y,z,Point,x,y,z,dx,dy,dz\n');
%
%*****************************************************************
%
% A Site Metrology System (SMS) was used to determine reference
% (x,y) values at 4 points on the top flange of each I-beam for
% each angular position on the floor. These points are initially
% stored in the coordinate system of the SMS but are converted to
% the coordinate system of the scanner. This was done using three
% reference spheres. A transformation was developed to convert the
% SMS coordinates to scanner coordinates. The SMS to scanner
% coordinate transformation is based on reference sphere
% centers. The transformation is given by:
%
% Scanner_coord = Transform*(sms_coord - cg) + cf
%
% The transformation is not restricted to rotation but includes a shift.
%
%*****************************************************************
cg = [10.13450; -2.051750; -0.3965000];
cf = [-2.655250; 4.259250; -0.2567500];
Transform = [0.4362677 0.8998131 0.0026162; -0.8998142 0.4362716 -0.0011596; -0.0021848 -0.0018482 0.9999958];
%***********************************************************
%
% Read in the database of I-beams and generate ideal I-beam
% points. These values need only be read once.
%
% The database values defined in inches to conform to usual
% Engineering practice.
%
% The database has an entry record for each I-beam. The records
% are structured as follows:
% Number of I-beams in the database
%
```

```
% For each I-beam
%
% Name of the I-beam
% Length of the I-beam
% Depth of the I-beam
% Flange Width
% Flange thickness
% Web thickness
%
%*************************************************************

fid_db = fopen('I_beam_database.txt','r');
no_beams = fscanf(fid_db,'%d',1);
%
% Create points that define the I-beams with the zero located at the center
% of mass, convert to meters by multiplying by 0.0254 in/m.
%
% Structures will be used to store and reference I-beam data
%
max_length = 0.0;
min_length = 1e6;
max_depth = 0.0;
min_depth = 1e6;
max_width = 0.0;
min_width = 1e6;
%
% Loop over all of the I-beams in the database
%
for i = 1:no_beams
   I_beam(i).name = fscanf(fid_db,'%s',1);
   I_beam(i).length = fscanf(fid_db,'%f',1);
   if (I_beam(i).length >= max_length)
      max_length = I_beam(i).length;
   end
   if (I_beam(i).length <= min_length)
      min_length = I_beam(i).length;
   end
   I_beam(i).depth = fscanf(fid_db,'%f',1);
   if (I_beam(i).depth >= max_depth)
      max_depth = I_beam(i).depth;
   end
   if (I_beam(i).depth <= min_depth)
      min_depth = I_beam(i).depth;
   end
   I_beam(i).flange_width = fscanf(fid_db,'%f',1);
   if (I_beam(i).flange_width >= max_width)
      max_width = I_beam(i).flange_width;
   end
   if (I_beam(i).flange_width <= min_width)
      min_width = I_beam(i).flange_width;
   end
   %
   %convert inches to meters
   %
   max_beam_length = max_length*0.0254;
   min_beam_length = min_length*0.0254;
```

```
max_beam_depth = max_depth*0.0254;
min_beam_depth = min_depth*0.0254;
max_beam_width = max_width*0.0254;
min_beam_width = min_width*0.0254;
I_beam(i).flange_thick = fscanf(fid_db,'%f',1);
I_beam(i).web_thick = fscanf(fid_db,'%f',1);
I_beam(i).web_height = I_beam(i).depth - 2*I_beam(i).flange_thick;
%
% Compute twenty four points describing each I-beam
%
I_beam(i).x(1) = 0.5*I_beam(i).flange_width;
I_beam(i).y(1) = 0.5*I_beam(i).length;
I_beam(i).z(1) = -0.5*I_beam(i).depth;
I_beam(i).x(2) = -0.5*I_beam(i).flange_width;
I_beam(i).y(2) = 0.5*I_beam(i).length;
I_beam(i).z(2) = -0.5*I_beam(i).depth;
I_beam(i).x(3) = 0.5*I_beam(i).flange_width;
I_beam(i).y(3) = 0.5*I_beam(i).length;
I_beam(i).z(3) = -0.5*I_beam(i).web_height;
I_beam(i).x(4) = -0.5*I_beam(i).flange_width;
I_beam(i).y(4) = 0.5*I_beam(i).length;
I_beam(i).z(4) = -0.5*I_beam(i).web_height;
I_beam(i).x(5) = 0.5*I_beam(i).web_thick;
I_beam(i).y(5) = 0.5*I_beam(i).length;
I_beam(i).z(5) = -0.5*I_beam(i).web_height;
I_beam(i).x(6) = -0.5*I_beam(i).web_thick;
I_beam(i).y(6) = 0.5*I_beam(i).length;
I_beam(i).z(6) = -0.5*I_beam(i).web_height;
I_beam(i).x(7) = 0.5*I_beam(i).web_thick;
I_beam(i).y(7) = 0.5*I_beam(i).length;
I_beam(i).z(7) = 0.5*I_beam(i).web_height;
I_beam(i).x(8) = -0.5*I_beam(i).web_thick;
I_beam(i).y(8) = 0.5*I_beam(i).length;
I_beam(i).z(8) = 0.5*I_beam(i).web_height;
I_beam(i).x(9) = 0.5*I_beam(i).flange_width;
I_beam(i).y(9) = 0.5*I_beam(i).length;
I_beam(i).z(9) = 0.5*I_beam(i).web_height;
I_beam(i).x(10) = -0.5*I_beam(i).flange_width;
I_beam(i).y(10) = 0.5*I_beam(i).length;
I_beam(i).z(10) = 0.5*I_beam(i).web_height;
I_beam(i).x(11) = 0.5*I_beam(i).flange_width;
I_beam(i).y(11) = 0.5*I_beam(i).length;
I_beam(i).z(11) = 0.5*I_beam(i).depth;
I_beam(i).x(12) = -0.5*I_beam(i).flange_width;
I_beam(i).y(12) = 0.5*I_beam(i).length;
I_beam(i).z(12) = 0.5*I_beam(i).depth;
I_beam(i).x(13) = 0.5*I_beam(i).flange_width;
I_beam(i).y(13) = -0.5*I_beam(i).length;
I_beam(i).z(13) = -0.5*I_beam(i).depth;
I_beam(i).x(14) = -0.5*I_beam(i).flange_width;
I_beam(i).y(14) = -0.5*I_beam(i).length;
I_beam(i).z(14) = -0.5*I_beam(i).depth;
I_beam(i).x(15) = 0.5*I_beam(i).flange_width;
I_beam(i).y(15) = -0.5*I_beam(i).length;
I_beam(i).z(15) = -0.5*I_beam(i).web_height;
I_beam(i).x(16) = -0.5*I_beam(i).flange_width;
```

```
I_beam(i).y(16) = -0.5*I_beam(i).length;
I_beam(i).z(16) = -0.5*I_beam(i).web_height;
I_beam(i).x(17) = 0.5*I_beam(i).web_thick;
I_beam(i).y(17) = -0.5*I_beam(i).length;
I_beam(i).z(17) = -0.5*I_beam(i).web_height;
I_beam(i).x(18) = -0.5*I_beam(i).web_thick;
I_beam(i).y(18) = -0.5*I_beam(i).length;
I_beam(i).z(18) = -0.5*I_beam(i).web_height;
I_beam(i).x(19) = 0.5*I_beam(i).web_thick;
I_beam(i).y(19) = -0.5*I_beam(i).length;
I_beam(i).z(19) = 0.5*I_beam(i).web_height;
I_beam(i).x(20) = -0.5*I_beam(i).web_thick;
I_beam(i).y(20) = -0.5*I_beam(i).length;
I_beam(i).z(20) = 0.5*I_beam(i).web_height;
I_beam(i).x(21) = 0.5*I_beam(i).flange_width;
I_beam(i).y(21) = -0.5*I_beam(i).length;
I_beam(i).z(21) = 0.5*I_beam(i).web_height;
I_beam(i).x(22) = -0.5*I_beam(i).flange_width;
I_beam(i).y(22) = -0.5*I_beam(i).length;
I_beam(i).z(22) = 0.5*I_beam(i).web_height;
I_beam(i).x(23) = 0.5*I_beam(i).flange_width;
I_beam(i).y(23) = -0.5*I_beam(i).length;
I_beam(i).z(23) = 0.5*I_beam(i).depth;
I_beam(i).x(24) = -0.5*I_beam(i).flange_width;
I_beam(i).y(24) = -0.5*I_beam(i).length;
I_beam(i).z(24) = 0.5*I_beam(i).depth;
%
% Convert the coordinates from inches to meters
%
I_beam(i).x(:) = I_beam(i).x(:)*0.0254;
I_beam(i).y(:) = I_beam(i).y(:)*0.0254;
I_beam(i).z(:) = I_beam(i).z(:)*0.0254;
nn_ib = 24;
xx_ib = I_beam(i).x;
yy_ib = I_beam(i).y;
zz_ib = I_beam(i).z;
%
% Get the center of mass and principle axes of each
% I-beam. This is the first step in creating a
% bounding box around each I-beam.
%
[cm_ib,pr_ax_ib] = princ_axes(nn_ib,xx_ib',yy_ib',zz_ib');
%
% Store the center of mass in the I-beam structure
%
I_beam(i).ctr_mass.x = cm_ib(1,1);
I_beam(i).ctr_mass.y = cm_ib(2,1);
I_beam(i).ctr_mass.z = cm_ib(3,1);
%
% Store the vectors representing the principal axes
% in the I-beam Structure. The R vector represents
% the vector along the longest exis, S the next
% longest, and T the third. In the case of I-beams
% R points along the I-beam length, S along the height,
% and T across the flange.
%
```

```
I_beam(i).R.x = pr_ax_ib(1,1);
I_beam(i).R.y = pr_ax_ib(2,1);
I_beam(i).R.z = pr_ax_ib(3,1);
I_beam(i).S.x = pr_ax_ib(1,2);
I_beam(i).S.y = pr_ax_ib(2,2);
I_beam(i).S.z = pr_ax_ib(3,2);
I_beam(i).T.x = pr_ax_ib(1,3);
I_beam(i).T.y = pr_ax_ib(2,3);
I_beam(i).T.z = pr_ax_ib(3,3);
%
% Compute the inner products of points and principle axes. This is done
% to find the maximum and minimum extents of the bounding box around
% the I-beam. Essentially all of the projetions of the data vectors
% describin the I-beam are examined for maximum and minimum length.
%
for np = 1:nn_ib
   R_dot_P(np) = I_beam(i).R.x * xx_ib(np) + I_beam(i).R.y * yy_ib(np)...
      + I_beam(i).R.z * zz_ib(np);
   S_dot_P(np) = I_beam(i).S.x * xx_ib(np) + I_beam(i).S.y * yy_ib(np)...
      + I_beam(i).S.z * zz_ib(np);
   T_dot_P(np) = I_beam(i).T.x * xx_ib(np) + I_beam(i).T.y * yy_ib(np)...
      + I_beam(i).T.z * zz_ib(np);
end
max_R_dot_P = max(R_dot_P); min_R_dot_P = min(R_dot_P);
max_S_dot_P = max(S_dot_P); min_S_dot_P = min(S_dot_P);
max_T_dot_P = max(T_dot_P); min_T_dot_P = min(T_dot_P);
%
% Compute the extent multipliers
%
aa = ( max_R_dot_P - min_R_dot_P)/2;
bb = ( max_S_dot_P - min_S_dot_P)/2;
cc = ( max_T_dot_P - min_T_dot_P)/2;
%
% Compute the bounding box center
%
I_beam(i).Q.x = cm_ib(1,1);
I_beam(i).Q.y = cm_ib(2,1);
I_beam(i).Q.z = cm_ib(3,1);
%
% Get the eight corners of the box relative to the box
% center.
%
xt = I_beam(i).Q.x + aa*I_beam(i).R.x...
   + bb*I_beam(i).S.x + cc*I_beam(i).T.x;
I_beam(i).corner(1).x1 =xt;
yt = I_beam(i).Q.y + aa*I_beam(i).R.y...
   + bb*I_beam(i).S.y + cc*I_beam(i).T.y;
I_beam(i).corner(1).y1 = yt;
zt = I_beam(i).Q.z + aa*I_beam(i).R.z...
   + bb*I_beam(i).S.z + cc*I_beam(i).T.z;
I_beam(i).corner(1).z1 = zt;
xt = I_beam(i).Q.x + aa*I_beam(i).R.x...
   + bb*I_beam(i).S.x - cc*I_beam(i).T.x;
I_beam(i).corner(2).x1 = xt;
yt = I_beam(i).Q.y + aa*I_beam(i).R.y...
   + bb*I_beam(i).S.y - cc*I_beam(i).T.y;
```

```
I_beam(i).corner(2).y1 = yt;
zt = I_beam(i).Q.z + aa*I_beam(i).R.z...
    + bb*I_beam(i).S.z - cc*I_beam(i).T.z;
I_beam(i).corner(2).z1 = zt;
xt = I_beam(i).Q.x + aa*I_beam(i).R.x...
    - bb*I_beam(i).S.x + cc*I_beam(i).T.x;
I_beam(i).corner(3).x1 = xt;
yt = I_beam(i).Q.y + aa*I_beam(i).R.y...
    - bb*I_beam(i).S.y + cc*I_beam(i).T.y;
I_beam(i).corner(3).y1 = yt;
zt = I_beam(i).Q.z + aa*I_beam(i).R.z...
    - bb*I_beam(i).S.z + cc*I_beam(i).T.z;
I_beam(i).corner(3).z1 = zt;
xt = I_beam(i).Q.x + aa*I_beam(i).R.x...
    - bb*I_beam(i).S.x - cc*I_beam(i).T.x;
I_beam(i).corner(4).x1 = xt;
yt = I_beam(i).Q.y + aa*I_beam(i).R.y...
    - bb*I_beam(i).S.y - cc*I_beam(i).T.y;
I_beam(i).corner(4).y1 = yt;
zt = I_beam(i).Q.z + aa*I_beam(i).R.z...
    - bb*I_beam(i).S.z - cc*I_beam(i).T.z;
I_beam(i).corner(4).z1 = zt;
xt = I_beam(i).Q.x - aa*I_beam(i).R.x...
    + bb*I_beam(i).S.x + cc*I_beam(i).T.x;
I_beam(i).corner(5).x1 = xt;
yt = I_beam(i).Q.y - aa*I_beam(i).R.y...
    + bb*I_beam(i).S.y + cc*I_beam(i).T.y;
I_beam(i).corner(5).y1 = yt;
zt = I_beam(i).Q.z - aa*I_beam(i).R.z...
    + bb*I_beam(i).S.z + cc*I_beam(i).T.z;
I_beam(i).corner(5).z1 = zt;
xt = I_beam(i).Q.x - aa*I_beam(i).R.x...
    + bb*I_beam(i).S.x - cc*I_beam(i).T.x;
I_beam(i).corner(6).x1 = xt;
yt = I_beam(i).Q.y - aa*I_beam(i).R.y...
    + bb*I_beam(i).S.y - cc*I_beam(i).T.y;
I_beam(i).corner(6).y1 = yt;
zt = I_beam(i).Q.z - aa*I_beam(i).R.z...
    + bb*I_beam(i).S.z - cc*I_beam(i).T.z;
I_beam(i).corner(6).z1 = zt;
xt = I_beam(i).Q.x - aa*I_beam(i).R.x...
    - bb*I_beam(i).S.x + cc*I_beam(i).T.x;
I_beam(i).corner(7).x1 = xt;
yt = I_beam(i).Q.y - aa*I_beam(i).R.y...
    - bb*I_beam(i).S.y + cc*I_beam(i).T.y;
I_beam(i).corner(7).y1 = yt;
zt = I_beam(i).Q.z - aa*I_beam(i).R.z...
    - bb*I_beam(i).S.z + cc*I_beam(i).T.z;
I_beam(i).corner(7).z1 = zt;
xt = I_beam(i).Q.x - aa*I_beam(i).R.x...
    - bb*I_beam(i).S.x - cc*I_beam(i).T.x;
I_beam(i).corner(8).x1 = xt;
yt = I_beam(i).Q.y - aa*I_beam(i).R.y...
    - bb*I_beam(i).S.y - cc*I_beam(i).T.y;
I_beam(i).corner(8).y1 = yt;
zt = I_beam(i).Q.z - aa*I_beam(i).R.z...
```

```
        - bb*I_beam(i).S.z - cc*I_beam(i).T.z;
    I_beam(i).corner(8).z1 = zt;
end
fclose(fid_db);
%*****************************************************************
%
% Load the SMS Flange Corner Point file. These are the reference
% measurements in the SMS coordinate system of the four corner
% points of the top flange for each of the different I-beams in
% the database. The file format consists of records with the
% fields: The first field is a number idntifying the beam, e.g. 1 or 2.
% The second field is the corner number of the points describing the
% I-beam. The third field is the rotation angle of the I-beam in the
% floor coordinate sysstem, e. g. 0, 30, 45, 60, 90. The third
% field. The fourth, fifth, and sixth fields are the x, y, z values
% of the corner point in the SMS coordinate frame.
%
%*****************************************************************
sms_file = 'SMS_I_beam_points.txt';
sms = load(sms_file);
[sm,sn] = size(sms);
beam_no = zeros(sm,1);
corner_no = zeros(sm,1);
rot_angle = zeros(sm,1);
sms_x = zeros(sm,1);
sms_y = zeros(sm,1);
sms_z = zeros(sm,1);
beam_no(:,1) = sms(:,1);
corner_no(:,1) = sms(:,2);
rot_angle(:,1) = sms(:,3);
sms_x(:,1) = sms(:,4);
sms_y(:,1) = sms(:,5);
sms_z(:,1) = sms(:,6);
%*************************************************************
%
% Open the file that has all of the names of the scan files.
% In a while loop open each scan file for reading.
% The Ladar data file is % assumed to be in a 3 column
% format of x, y, z data values in meters.
%
%*****************************************************
fid1 = fopen('All_scanfile_files_1.txt','r');
while feof(fid1)==0 %see end for end statement
    % clear variables before reading each scan file since the sysstem may
    % retain undesirable values and produce unexpected results.
    clear ladar_file max_x min_x max_y min_y max_z min_z ncellx ncelly ncellz
    clear del_cellx del_celly del_cellz cell_x dx cell_y dy cell_z dz bin_count
    clear nz_i_bins nz_j_binx nz_k_bins count lnz nz_i_bins_x nz_j_bins_y nz_k_bins_z
    clear Max_string_length Max_L nz_i_bins_new nz_j_bins_new nz_k_bins_new
    clear nz_i_bins_x_new nz_j_bins_y_new nz_k_bins_z_new potential_object obj_no
    clear n_obj object R_dot_P S_dot_P T_dot_P max_R_dot_P min_R_dot_P max_S_dot_P
    clear min_S_dot_P max_T_dot_P min_T_dot_P aa bb cc cc1 AA rt Rot T H mpt trmpt
    clear comparison_error min_error i_min obj_min
    ladar_file = fgetl(fid1); % read the scan file name
    disp(sprintf('Processing file = %s\n',ladar_file));
    a = load(ladar_file); % load the scan file data into a matrix
```

```
    %strip off the .txt extension and use clear( ) to clear variable
    %with string name
    place = findstr(ladar_file,'.txt');
    ladar_file_minus_txt = ladar_file(1:place-1);
    clear(ladar_file_minus_txt);
    [m,n] = size(a);
    xtemp = zeros(m,1);
    ytemp = zeros(m,1);
    ztemp = zeros(m,1);
    xtemp(:,1) = a(:,1);
    ytemp(:,1) = a(:,2);
    ztemp(:,1) = a(:,3);
    x = xtemp; % store the x, y, z scan coordinates in the scanner
    y = ytemp; % coordinate frame. The units are in meters.
    z = ztemp;
    max_x = max(x);
    min_x = min(x);
    max_y = max(y);
    min_y = min(y);
    max_z = max(z);
    min_z = min(z);
%**********************************************************
%
% In preparation for binning of data get the number of cells or
% voxels that will be used to bin the scan data points.
% Get the number of cells in each direction. 60x60x60 is a good
% compromise for the I-beam data sets.
%
%**********************************************************
    ncellx = 60;
    ncelly = 60;
    ncellz = 60;
    del_cellx = abs(max_x - min_x);
    del_celly = abs(max_y - min_y);
    del_cellz = abs(max_z - min_z);
%**********************************************************
%
% Bin the data into ncellx x ncelly x ncellz voxels. All the
% binned data in one voxel will be identified by the center
% of the voxel.
%
%**********************************************************

    [cell_x,dx,cell_y,dy,cell_z,dz,bin_count] = bin3d(ncellx,ncelly,ncellz,x,y,z);

%**********************************************************************
%
% Get indices and center coordinates of the voxels with bin counts > 1.
% Also get the voxel x, y, z center points.
% This is a first step to reducing the phantom pixels.
%
%**********************************************************************
%
    ind = 1;
    for i = 1:ncellx
        for j = 1:ncelly
```

```
      for k = 1:ncellz
         if (bin_count(i,j,k) > 1)
            nz_i_bins(ind) = i;
            nz_j_bins(ind) = j;
            nz_k_bins(ind) = k;
            count(ind) = bin_count(i,j,k);
            ind = ind +1;
         else
            bin_count(i,j,k) = 0;
         end
      end
   end
end
lnz = length(nz_i_bins);
for r = 1:lnz
   nz_i_bins_x(r) = cell_x(nz_i_bins(r));
   nz_j_bins_y(r) = cell_y(nz_j_bins(r));
   nz_k_bins_z(r) = cell_z(nz_k_bins(r));
end
%***************************************************************************
%
% Compute string lengths up the columns of non-zero bin count voxels.
% The strategy here involves grouping or concatenating voxels together
% so that these groups can be compared against some minimal length related
% to the height of the I-beam.
%
%***************************************************************************

[Max_string_length,Max_L]=get_vertical_string_length(ncellx,ncelly,ncellz,bin_count);

%***************************************************************************
%
% If the max length of nonzero entries in a column is less than a
% specified amount make the column all zero. The product of the maximum
% string length and the voxel height must be less than the minmum depth
% of the I-beams for the column to be zeroed.
%
%***************************************************************************
for i = 1:ncellx
   for j = 1:ncelly
      if (Max_L(i,j)*del_cellz < min_depth)
         bin_count(i,j,:) = 0;
      end
   end
end
%***************************************************************************
%
% Recompute max lengths lengths.
%
%***************************************************************************

[Max_string_length,Max_L]=get_vertical_string_length(ncellx,ncelly,ncellz,bin_count);

%**************************************************************************
%
% At this point we begin the object segmentation algorithm.
```

```
% The segmentation algorithm works with an associated array called a
% mask array that has the same dimension os the number of voxels along each
% axis. The mask array has entrie of 1 if the associated voxel has data
% points within and 0 otherwise. The use of this mask array and the
% segmentation algorithm are described in the function Segment_objects.
%
%*************************************************************************
   mask = zeros(ncellx,ncelly,ncellz);
   ind = 1;
   for i = 1:ncellx
      for j = 1:ncelly
         for k = 1:ncellz
            if (bin_count(i,j,k) > 0)
               nz_i_bins_new(ind) = i;
               nz_j_bins_new(ind) = j;
               nz_k_bins_new(ind) = k;
               ind = ind +1;
               mask(i,j,k) = 1;
            end
         end
      end
   end
   tot_nz_bins = ind-1;
%*************************************************************************
%
% Asociate x, y, z values with these bins
%
%*************************************************************************
   lnz1 = length(nz_i_bins_new);
   nz_i_bins_x_new = zeros(lnz1);

   nz_j_bins_y_new = zeros(lnz1);
   nz_k_bins_z_new = zeros(lnz1);
   for r = 1:lnz1
      nz_i_bins_x_new(r) = cell_x(nz_i_bins_new(r));
      nz_j_bins_y_new(r) = cell_y(nz_j_bins_new(r));
      nz_k_bins_z_new(r) = cell_z(nz_k_bins_new(r));
   end
%*************************************************************************
%
% This section segments all potential objects. They are segmented using
% the mask array. After all potential objects are found a matrix test is
% performed to determine if an object is a significant 3D object. Any
% objects returned with only 1 (point), 2 (line), or 3 (plane) points will
% be immediately deleted from consideration as legitimate objects. For
% objects with more that 3 points tests have to be performed.
%
%potential_object is an array of sructures defined as follow:
%potential_object(obj_no).size  is the number of voxels in the object = count
%potential_object(obj_no).index(count,1) = ii; i index for current voxel
%potential_object(obj_no).index(count,2) = jj; j index for current voxel
%potential_object(obj_no).index(count,3) = kk; k index for current voxel
%potential_object(obj_no).coord(count,1) = cell_x(ii); center x for current voxel
%potential_object(obj_no).coord(count,2) = cell_y(jj); center y for current voxel
%potential_object(obj_n0).coord(count,3) = cell_z(kk); ccenter z for current voxel
%*************************************************************************
```

```
   [obj_no,
potential_object]=Segment_objects(lnz1,nz_i_bins_new,nz_j_bins_new,nz_k_bins_new,mask,cell_x,cell_
y,cell_z,ncellx,ncelly,ncellz);

%**************************************************************************
%
% Select objects that have three dimensions to them otherwise the bounding
% boxes are not constructable.
%
%***************************************************************************

   [n_obj,object]=Object_test(obj_no,potential_object);

%**************************************************************************
%
% Build bounding boxes around objects. Loop over the objects. The algorithm
% here is the same as that used to create bounding boxes around the ideal
% I-beams in the database.
%
%**************************************************************************

   for obj = 1:n_obj
      nn = object(obj).size;
      coord = zeros(nn,3);
      x_coord = zeros(nn,1);
      y_coord = zeros(nn,1);
      z_coord = zeros(nn,1);
      coord = object(obj).coord;
      x_coord(:,1) = coord(:,1);
      y_coord(:,1) = coord(:,2);
      z_coord(:,1) = coord(:,3);
      [cm,pr_ax] = princ_axes(nn,x_coord,y_coord,z_coord); %get center of mass and principle axes
      object(obj).ctr_mass.x = cm(1,1);
      object(obj).ctr_mass.y = cm(2,1);
      object(obj).ctr_mass.z = cm(3,1);
      object(obj).R.x = pr_ax(1,1);
      object(obj).R.y = pr_ax(2,1);
      object(obj).R.z = pr_ax(3,1);
      object(obj).S.x = pr_ax(1,2);
      object(obj).S.y = pr_ax(2,2);
      object(obj).S.z = pr_ax(3,2);
      object(obj).T.x = pr_ax(1,3);
      object(obj).T.y = pr_ax(2,3);
      object(obj).T.z = pr_ax(3,3);
      for np = 1:nn %compute inner products of points to determine principle axes extents
         R_dot_P(np) = object(obj).R.x * x_coord(np) + object(obj).R.y * y_coord(np)...
            + object(obj).R.z * z_coord(np);
         S_dot_P(np) = object(obj).S.x * x_coord(np) + object(obj).S.y * y_coord(np)...
            + object(obj).S.z * z_coord(np);
         T_dot_P(np) = object(obj).T.x * x_coord(np) + object(obj).T.y * y_coord(np)...
            + object(obj).T.z * z_coord(np);
      end
      max_R_dot_P = max(R_dot_P); min_R_dot_P = min(R_dot_P);
      max_S_dot_P = max(S_dot_P); min_S_dot_P = min(S_dot_P);
      max_T_dot_P = max(T_dot_P); min_T_dot_P = min(T_dot_P);
```

```
  aa = ( max_R_dot_P - min_R_dot_P)/2;
  bb = ( max_S_dot_P - min_S_dot_P)/2;
  cc = ( max_T_dot_P - min_T_dot_P)/2;
% Compute the bounding box center
  object(obj).Q.x = cm(1,1);
  object(obj).Q.y = cm(2,1);
  object(obj).Q.z = cm(3,1);
% Get the eight corners of the box.
  object(obj).corner(1).x = object(obj).Q.x + aa*object(obj).R.x...
    + bb*object(obj).S.x + cc*object(obj).T.x;
  object(obj).corner(1).y = object(obj).Q.y + aa*object(obj).R.y...
    + bb*object(obj).S.y + cc*object(obj).T.y;
  object(obj).corner(1).z = object(obj).Q.z + aa*object(obj).R.z...
    + bb*object(obj).S.z + cc*object(obj).T.z;
  object(obj).corner(2).x = object(obj).Q.x + aa*object(obj).R.x...
    + bb*object(obj).S.x - cc*object(obj).T.x;
  object(obj).corner(2).y = object(obj).Q.y + aa*object(obj).R.y...
    + bb*object(obj).S.y - cc*object(obj).T.y;
  object(obj).corner(2).z = object(obj).Q.z + aa*object(obj).R.z...
    + bb*object(obj).S.z - cc*object(obj).T.z;
  object(obj).corner(3).x = object(obj).Q.x + aa*object(obj).R.x...
    - bb*object(obj).S.x + cc*object(obj).T.x;
  object(obj).corner(3).y = object(obj).Q.y + aa*object(obj).R.y...
    - bb*object(obj).S.y + cc*object(obj).T.y;
  object(obj).corner(3).z = object(obj).Q.z + aa*object(obj).R.z...
    - bb*object(obj).S.z + cc*object(obj).T.z;
  object(obj).corner(4).x = object(obj).Q.x + aa*object(obj).R.x...
    - bb*object(obj).S.x - cc*object(obj).T.x;
  object(obj).corner(4).y = object(obj).Q.y + aa*object(obj).R.y...
    - bb*object(obj).S.y - cc*object(obj).T.y;
  object(obj).corner(4).z = object(obj).Q.z + aa*object(obj).R.z...
    - bb*object(obj).S.z - cc*object(obj).T.z;
  object(obj).corner(5).x = object(obj).Q.x - aa*object(obj).R.x...
    + bb*object(obj).S.x + cc*object(obj).T.x;
  object(obj).corner(5).y = object(obj).Q.y - aa*object(obj).R.y...
    + bb*object(obj).S.y + cc*object(obj).T.y;
  object(obj).corner(5).z = object(obj).Q.z - aa*object(obj).R.z...
    + bb*object(obj).S.z + cc*object(obj).T.z;
  object(obj).corner(6).x = object(obj).Q.x - aa*object(obj).R.x...
    + bb*object(obj).S.x - cc*object(obj).T.x;
  object(obj).corner(6).y = object(obj).Q.y - aa*object(obj).R.y...
    + bb*object(obj).S.y - cc*object(obj).T.y;
  object(obj).corner(6).z = object(obj).Q.z - aa*object(obj).R.z...
    + bb*object(obj).S.z - cc*object(obj).T.z;
  object(obj).corner(7).x = object(obj).Q.x - aa*object(obj).R.x...
    - bb*object(obj).S.x + cc*object(obj).T.x;
  object(obj).corner(7).y = object(obj).Q.y - aa*object(obj).R.y...
    - bb*object(obj).S.y + cc*object(obj).T.y;
  object(obj).corner(7).z = object(obj).Q.z - aa*object(obj).R.z...
    - bb*object(obj).S.z + cc*object(obj).T.z;
  object(obj).corner(8).x = object(obj).Q.x - aa*object(obj).R.x...
    - bb*object(obj).S.x - cc*object(obj).T.x;
  object(obj).corner(8).y = object(obj).Q.y - aa*object(obj).R.y...
    - bb*object(obj).S.y - cc*object(obj).T.y;
  object(obj).corner(8).z = object(obj).Q.z - aa*object(obj).R.z...
    - bb*object(obj).S.z - cc*object(obj).T.z;
```

```
   end
%************************************************************************
%
% Now compare the bounding boxes of the true objects with the
% ideal bounding boxes of the database I-beams. The outer loop indexes
% the ideal I-beam models.
%
%************************************************************************
   for i = 1:no_beams
      for obj = 1:n_obj
      %Next find the rotation that aligns the principal axes
      %of the model I_beam box with the object box
         cc1 = zeros(9,1);
         %set up the right hand side of the object eigenvectors
         cc1(1,1) = object(obj).R.x;
         cc1(2,1) = object(obj).R.y;
         cc1(3,1) = object(obj).R.z;
         cc1(4,1) = object(obj).S.x;
         cc1(5,1) = object(obj).S.y;
         cc1(6,1) = object(obj).S.z;
         cc1(7,1) = object(obj).T.x;
         cc1(8,1) = object(obj).T.y;
         cc1(9,1) = object(obj).T.z;
         %set up the 9 x 9 matrix of model eigenvectors
         AA = zeros(9,9);
         AA(1,1) = I_beam(i).R.x;
         AA(1,2) = I_beam(i).R.y;
         AA(1,3) = I_beam(i).R.z;
         AA(2,4) = I_beam(i).R.x;
         AA(2,5) = I_beam(i).R.y;
         AA(2,6) = I_beam(i).R.z;
         AA(3,7) = I_beam(i).R.x;
         AA(3,8) = I_beam(i).R.y;
         AA(3,9) = I_beam(i).R.z;
         AA(4,1) = I_beam(i).S.x;
         AA(4,2) = I_beam(i).S.y;
         AA(4,3) = I_beam(i).S.z;
         AA(5,4) = I_beam(i).S.x;
         AA(5,5) = I_beam(i).S.y;
         AA(5,6) = I_beam(i).S.z;
         AA(6,7) = I_beam(i).S.x;
         AA(6,8) = I_beam(i).S.y;
         AA(6,9) = I_beam(i).S.z;
         AA(7,1) = I_beam(i).T.x;
         AA(7,2) = I_beam(i).T.y;
         AA(7,3) = I_beam(i).T.z;
         AA(8,4) = I_beam(i).T.x;
         AA(8,5) = I_beam(i).T.y;
         AA(8,6) = I_beam(i).T.z;
         AA(9,7) = I_beam(i).T.x;
         AA(9,8) = I_beam(i).T.y;
         AA(9,9) = I_beam(i).T.z;
         %solve least squares for the rotation matrix
         rt = zeros(9,1);
         Rot = zeros(3,3);
         rt = AA\cc1;
```

```
            %Make the rotation matrix a 2D array
            for ii = 1:3
               for jj = 1:3
                  kk = (ii-1)*3 + jj;
                  Rot(ii,jj) = rt(kk,1);
               end
            end
            %Apply the translation and rotation to the model data with a homogeneous
            %transformation
            T(1,1) = object(obj).ctr_mass.x;
            T(2,1) = object(obj).ctr_mass.y;
            T(3,1) = object(obj).ctr_mass.z;
            %Set up homogeneous transformation
            H = zeros(4,4);
            for ii = 1:3
               for jj = 1:3
                  H(ii,jj) = Rot(ii,jj);
               end
            end
            H(1,4) = T(1,1);
            H(2,4) = T(2,1);
            H(3,4) = T(3,1);
            H(4,4) = 1.0;
            %Rotate and translate the bounding box for each beam model
            %Compute the sum of squares differences between the true object
            %and the beam
            error = 0;
            for v = 1:8
               mpt(1,1) = I_beam(i).corner(v).x1;
               mpt(2,1) = I_beam(i).corner(v).y1;
               mpt(3,1) = I_beam(i).corner(v).z1;
               mpt(4,1) = 1.0;
               trmpt = H*mpt; %apply homogeneous transformation
               I_beam(i).corner(v).trrot(obj,1).x2 = trmpt(1,1);
               I_beam(i).corner(v).trrot(obj,1).y2 = trmpt(2,1);
               I_beam(i).corner(v).trrot(obj,1).z2 = trmpt(3,1);
               error = error + (object(obj).corner(v).x -...
                  I_beam(i).corner(v).trrot(obj,1).x2)^2 + (object(obj).corner(v).y -...
                  I_beam(i).corner(v).trrot(obj,1).y2)^2 + (object(obj).corner(v).z -...
                  I_beam(i).corner(v).trrot(obj,1).z2)^2;
               comparison_error(i,obj) = sqrt(error);
            end
         end
      end
%first of all find the true object and I-beam model that produces the
%minimum error. Initialize min_error to a large number
   min_error = 1e+10;
   for i = 1:no_beams
      for obj = 1:n_obj
         disp(sprintf('beam = %d obj = %d comp_err = %f\n',i,obj,comparison_error(i,obj)));
         if (comparison_error(i,obj) < min_error)
            min_error = comparison_error(i,obj);
            i_min = i;
            obj_min = obj;
         end
      end
```

```
   end
   i_min
   obj_min
   I_beam(i_min).name
%*************************************************************************
% Compute the pose angle of the object relative to the scanner
%*************************************************************************
pose_angle = zeros(n_obj,1);
pose_angle(obj_min) = (180/pi)* atan(object(obj_min).R.y/object(obj_min).R.x);
if (pose_angle(obj_min) < 0)
   pose_angle(obj_min) = 180 + pose_angle(obj_min);
end
%*************************************************************************
% Section to compare bounding box points with reference points
% Reading the sms measurements for the four top flange corners
% sms point a = bounding box corner 5
% sms point b = bounding box corner 6
% sms point c = bounding box corner 1
% sms point d = bounding box corner 2
%*************************************************************************
% Getting table entry for sms point a (bounding box corner 5)
if((ladar_file(1) =='A')& (ladar_file(2) == '0'))

   entry = 1;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '3') & (ladar_file(3) == '0'))
   entry = 5;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '4') & (ladar_file(3) == '5'))
    entry = 9;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '6') & (ladar_file(3) == '0'))
    entry = 13;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '9') & (ladar_file(3) == '0'))
    entry = 17;
elseif((ladar_file(1) =='B')& (ladar_file(2) == '0'))
    entry = 21;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '3') & (ladar_file(3) == '0'))
    entry = 25;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '4') & (ladar_file(3) == '5'))
    entry = 29;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '6') & (ladar_file(3) == '0'))
    entry = 33;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '9') & (ladar_file(3) == '0'))
    entry = 37;
end
%*************************************************************************
%
% Put sms points into scanner frame for comparison. File name provides the
% entry point into the sms corner measurement file.
% At this point we need to determine which points to compare
% The four sms points are defined as a, b, c, and d. The four top bounding
% box points are 1,2, 5,and 6. We either (1) compare a to 1, b to 2, c to 5,
% and d to 6 or (2) we compare a to 5, b to 6, c to 1, and d to 2
% To determine which set of comparisons to make we will generate two unit
% vectors, one between sms points c to a and one from bounding box point 5
% to 1. We then take the inner product. If it is negative then we do
% comparison (2) otherwise (1).
%
```

```
%**************************************************************************
scanner_pt = zeros(3,1);
sms_pt = zeros(3,1);
ca = [sms_x(entry,1)-sms_x(entry+2,1);sms_y(entry,1)-sms_y(entry+2,1)];
normca = norm(ca);
cau = ca/normca;
v51 = [I_beam(i_min).corner(1).trrot(obj_min,1).x2-
I_beam(i_min).corner(5).trrot(obj_min,1).x2;I_beam(i_min).corner(1).trrot(obj_min,1).y2-
I_beam(i_min).corner(5).trrot(obj_min,1).y2];
norm51 = norm(v51);
v51u = v51/norm51;
inner_prod = cau(1)*v51u(1) + cau(2)*v51u(2);
if (inner_prod < 0)
   index(1) = 5;
   index(2) = 6;
   index(3) = 1;
   index(4) = 2;
else
   index(1) = 1;
   index(2) = 2;
   index(3) = 5;
   index(4) = 6;
end
%generate a line of blank cells
fprintf(fid3,' , , , , , , , , , , , , , , , , ,  \n');
% sms point a to scanner point index(1) comparison. First transform to scanner frame.
sms_pt(1,1) = sms_x(entry,1);
sms_pt(2,1) = sms_y(entry,1);
sms_pt(3,1) = sms_z(entry,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) - I_beam(i_min).corner(index(1)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) - I_beam(i_min).corner(index(1)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) - I_beam(i_min).corner(index(1)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f , %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min).ctr_mass.x,object(obj_m
in).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'a',scanner_pt(1,1),scanner_pt(2,1),scanner_pt(3,1),inde
x(1),...

I_beam(i_min).corner(index(1)).trrot(obj_min,1).x2,I_beam(i_min).corner(index(1)).trrot(obj_min,1).y2,...
   I_beam(i_min).corner(index(1)).trrot(obj_min,1).z2,error_x,error_y,error_z);
% sms point b to scanner point index(2) comparison. First transform to scanner frame.
sms_pt(1,1) = sms_x(entry+1,1);
sms_pt(2,1) = sms_y(entry+1,1);
sms_pt(3,1) = sms_z(entry+1,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) - I_beam(i_min).corner(index(2)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) - I_beam(i_min).corner(index(2)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) - I_beam(i_min).corner(index(2)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f , %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min).ctr_mass.x,object(obj_m
in).ctr_mass.y,...
```

```
object(obj_min).ctr_mass.z,pose_angle(obj_min),'b',scanner_pt(1,1),scanner_pt(2,1),scanner_pt(3,1),inde
x(2),...

I_beam(i_min).corner(index(2)).trrot(obj_min,1).x2,I_beam(i_min).corner(index(2)).trrot(obj_min,1).y2,...
   I_beam(i_min).corner(index(2)).trrot(obj_min,1).z2,error_x,error_y,error_z);
% sms point c to scanner point index(3) comparison. First transform to scanner frame.
sms_pt(1,1) = sms_x(entry+2,1);
sms_pt(2,1) = sms_y(entry+2,1);
sms_pt(3,1) = sms_z(entry+2,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) - I_beam(i_min).corner(index(3)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) - I_beam(i_min).corner(index(3)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) - I_beam(i_min).corner(index(3)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f , %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min).ctr_mass.x,object(obj_m
in).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'c',scanner_pt(1,1),scanner_pt(2,1),scanner_pt(3,1),inde
x(3),...

I_beam(i_min).corner(index(3)).trrot(obj_min,1).x2,I_beam(i_min).corner(index(3)).trrot(obj_min,1).y2,...
   I_beam(i_min).corner(index(3)).trrot(obj_min,1).z2,error_x,error_y,error_z);
% sms point d to scanner point index(4) comparison. First transform to scanner frame.
sms_pt(1,1) = sms_x(entry+3,1);
sms_pt(2,1) = sms_y(entry+3,1);
sms_pt(3,1) = sms_z(entry+3,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) - I_beam(i_min).corner(index(4)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) - I_beam(i_min).corner(index(4)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) - I_beam(i_min).corner(index(4)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f , %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min).ctr_mass.x,object(obj_m
in).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'d',scanner_pt(1,1),scanner_pt(2,1),scanner_pt(3,1),inde
x(4),...

I_beam(i_min).corner(index(4)).trrot(obj_min,1).x2,I_beam(i_min).corner(index(4)).trrot(obj_min,1).y2,...
   I_beam(i_min).corner(index(4)).trrot(obj_min,1).z2,error_x,error_y,error_z);


end
fclose(fid1);
fclose(fid3);
% End the main segmentation algorithm by binning
```

A.1.1.2 BIN3D SCRIPT

```
function [cell_x,dx,cell_y,dy,cell_z,dz,bin_count] = bin3d(ncellx,ncelly,ncellz,x,y,z)
%BIN3D - bins a set of x, y, z points into a 3d cellarray of siz ncellx x
```

```
%ncelly x ncellz. The algorithm proceeds by getting the integer indices of
%the 3d voxel to which each point belongs.It then conts numbers of index
%matches. The function also returns the center coordinates of each of the
%voxels in cell_x, cell_y, cellz.

%Initialize arrays
Npts = length(x);
Ix = zeros(Npts,1);
Iy = zeros(Npts,1);
Iz = zeros(Npts,1);
bin_count=zeros(ncellx,ncelly,ncellz);

%get voxel dimensions
maxx=max(x);minx=min(x);
maxy=max(y);miny=min(y);
maxz=max(z);minz=min(z);
dx = (maxx - minx)/ncellx;
dy = (maxy - miny)/ncelly;
dz = (maxz - minz)/ncellz;

%get vectors of cell indices
%fix up boundary cells
for r = 1:Npts
   Ix(r) = fix((x(r)-minx)/dx)+1;
   if (Ix(r)==(ncelly+1))
            Ix(r) = ncellx;
   end
   Iy(r) = fix((y(r)-miny)/dy)+1;
   if (Iy(r)==(ncelly+1))
            Iy(r) = ncelly;
   end
   Iz(r) = fix((z(r)-minz)/dz)+1;
   if (Iz(r)==(ncellz+1))
            Iz(r) = ncellz;
   end
end

%Get the bin counts by finding matches, counting them, and deleting them
%from the indes lists
remainpts = Npts;
while 0<remainpts
         %begins with current first pt. in each array.
         Inx = Ix(1);
   Iny = Iy(1);
   Inz = Iz(1);
         k = find((Ix==Inx)&(Iy==Iny)&(Iz==Inz));
         lgthk = length(k);
         bin_count(Inx,Iny,Inz) = lgthk;
         %eliminate already counted elements
         Ix(k) = [];
         Iy(k) = [];
         Iz(k) = [];
         %compute remaining points in these arrays
         remainpts = remainpts - lgthk;
         %go back with reduced arrays
end
```

```
%compute voxel center coordinates
cell_x = zeros(ncellx,1);
cell_y = zeros(ncelly,1);
cell_z = zeros(ncellz,1);
cell_x(1) = minx + dx/2;
for i = 2:ncellx
   cell_x(i) = cell_x(i-1) + dx;
end
cell_y(1) = miny + dy/2;
for i = 2:ncelly
   cell_y(i) = cell_y(i-1) + dy;
end
cell_z(1) = minz + dz/2;
for i = 2:ncellz
   cell_z(i) = cell_z(i-1) + dz;
end
```

A.1.1.3 PRINC_AXES SCRIPT

```
function [cm,p_axis] = princ_axes(n,x,y,z)
%PRINC_AXES - This function produces the principal axes of a set of n points
%given as x, y, z vectors. The method used is principal components
%analysis. The method is given in the text
%
% E. Lengyel, "Mathematics for 3D Game Programming & Computer Graphics",
% Charles River Media, Inc. Hingham, Massachusetts, 2001
%
%The function returns the center of mass in the vector cm, where the first
%component is the x component, the second the y and the third the z. The
%3 x 3 array p_axis has the principal axis vectors arranged so that the
%first column is the first principal axis (longest dimension of a bounding
%box), the other two columns have the other axes directions in descending
%order of eigenvalues.

%First, compute the center of mass
sx = 0; sy = 0; sz = 0;
n = length(x);
for r = 1:n
   sx = sx + x(r,1);
   sy = sy + y(r,1);
   sz = sz + z(r,1);
end
cm = zeros(3,1);
cm(1,1) = sx/n;
cm(2,1) = sy/n;
cm(3,1) = sz/n;

%Form the Covariance Matrix
cv = zeros(3,3);
for r = 1:n
   a(1,1) = x(r,1) - cm(1,1);
   a(2,1) = y(r,1) - cm(2,1);
   a(3,1) = z(r,1) - cm(3,1);
```

```
    av = [a(1,1);a(2,1);a(3,1)];
    cv = cv + av*av';
end
cv = cv/n;

%The principal axes are associated with the eigenvectors of the
%covariance matrix, which is real symmetric in this case.
%The eigenvalues are real and in general distinct. The function
%eig returns a diagonal matrix eig_val of eigenvalues and a
%matrix of eigenvectors associated with the eigenvalues.

[eig_vect,eig_val] = eig(cv);

%In the current case eig_val is a 3 x 3 diagonal matrix and
%eig_vect is a 3 x 3 matrix with each column an eigenvector
%associated with the appropriate eigenvalue in eig_val. The eigenvectors
%are normalized to 1 in the Euclidean norm. The eigenvectors are the
%principal axis directions ordered by the eigenvalues. That is, the first
%principal axis is associated with the largest eigenvalue, etc. To return
%the eigenvectors in the order of the principal axes we sort the
%eigenvalues using sort which produces an ascending order sort. We have to
%reverse the associated eigenvectors on return.

eg = [eig_val(1,1);eig_val(2,2);eig_val(3,3)];
[eigv,I] = sort(eg);

%This produces eigv(I(1)) < eigv(I(2)) < eigv(I(3))
%The associated eigenvectors are eig_vect(:,I(1)), etc. Return the
%eigenvectors in a matrix in reverse order.

p_axis = zeros(3,3);
p_axis(:,1) = eig_vect(:,I(3));
p_axis(:,2) = eig_vect(:,I(2));
p_axis(:,3) = eig_vect(:,I(1));
```

A.1.1.4 GET_VERTICAL_STRING_LENGTH.M

```
function [Max_string_length,Max_L]=get_vertical_string_length(ncellx,ncelly,ncellz,bin_count)
%GET_VERTICAL_STRING_LENGTH - A vertical string of voxels is defined to be
%those voxels with non-zero bin count that are direct neighbors of each
%other. This function finds the maximum string lengths in each column of
%voxels as well as the overall maximum string length.
Max_L = zeros(ncellx,ncelly);
for i = 1:ncellx
    for j = 1:ncelly
        buf = zeros(ncellz,1);
        for k = 1:ncellz
            if(bin_count(i,j,k) > 0)
            buf(k) = 1;
            end
        end
        max_length = 0;
        string_length = 0;
        for k = 1:ncellz
```

```
            if (buf(k) == 1)
                string_length = string_length + 1;
                if (string_length >= max_length)
                    max_length = string_length;
                end
            else
                string_length = 0;
            end
        end
        Max_L(i,j) = max_length;
    end
end
Max_string_length = max(max(Max_L));
```

A.1.1.5 SEGMENT_OBJECTS.M

```
function [obj_no,
object]=Segment_objects(lnz1,nz_i_bins_new,nz_j_bins_new,nz_k_bins_new,mask,cell_x,cell_y,cell_z,n
cellx,ncelly,ncellz)
%************************************************************************
%Here we segment the objects in the filtered image and put bounding
%boxes around them. We will compare the box sizes against the I-beam
%specifications
%
%The basic idea of the object segmentation algorithm is to accumulate
%neighboring voxels into an object structure. The technique involves
%zeroing out the mask array for every neighboring point that is found
%and put on a stack. When there are no more nonzero neighboring mask
%elements to be zeroed out the stack is unloaded into an object structure.
%The algorithm then goes to the next object.
%
%Functions called: push, pop
%
%Ref: Nikolaidis, N., Pita, I., 3-D Image Processing Algorithms, John Wiley
% & Sons, Inc., New York, 2001.
%
%************************************************************************
%Initialize the stack
stack(1).i(1,1) = 0;
stack(1).j(1,1) = 0;
stack(1).k(1,1) = 0;
stack(1).x(1,1) = 0.0;
stack(1).y(1,1) = 0.0;
stack(1).z(1,1) = 0.0;
obj_no = 0; %Initialize the object number
for nz_vox = 1:lnz1
    ii = nz_i_bins_new(nz_vox);
    jj = nz_j_bins_new(nz_vox);
    kk = nz_k_bins_new(nz_vox);
    if(mask(ii,jj,kk) > 0)   %find the next voxel with non-zero mask value
        obj_no = obj_no + 1; %Create a new object
        mask(ii,jj,kk) = 0; %Zero out the current object seed point
        object(obj_no).size = 1; %Initialize the object structure
        object(obj_no).index(1,1) = ii;
```

```
object(obj_no).index(1,2) = jj;
object(obj_no).index(1,3) = kk;
object(obj_no).coord(1,1) = cell_x(ii);
object(obj_no).coord(1,2) = cell_y(jj);
object(obj_no).coord(1,3) = cell_z(kk);
cur_stack_top = 0; %Set the current stack top and push the first element onto the stack
[new_stack_top,stack] = push(ii,jj,kk,cell_x(ii),cell_y(jj),cell_z(kk),cur_stack_top,stack);
cur_stack_top = new_stack_top; %Update the stack top pointer
first_step = 1;
%step into a loop that builds the object from neighborhood voxels
%by expanded search using a LIFO stack to store voxel hits
while (cur_stack_top > 0)
   new_stack_top = pop(cur_stack_top);
   if (first_step > 0)
      object(obj_no).size = 1;
      count = 1; %This counts number of voxels in object
      first_step = 0;
   else
      object(obj_no).size = object(obj_no).size + 1;
      count = count + 1;
   end
   %unroll the stack and put them into the object
   i_cur = stack(cur_stack_top).i(1,1);
   object(obj_no).index(count,1) = i_cur;
   j_cur = stack(cur_stack_top).j(1,1);
   object(obj_no).index(count,2) = j_cur;
   k_cur = stack(cur_stack_top).k(1,1);
   object(obj_no).index(count,3) = k_cur;
   object(obj_no).coord(count,1) = stack(cur_stack_top).x(1,1);
   object(obj_no).coord(count,2) = stack(cur_stack_top).y(1,1);
   object(obj_no).coord(count,3) = stack(cur_stack_top).z(1,1);
   cur_stack_top = new_stack_top;
   %put neighbor nonzero voxels onto the stack
   for p = -1:1
      for q = -1:1
         for r = -1:1
            if((p~=0)|(q~=0)|(r~=0)) %skip center it has been stored above
               f = i_cur + p;
               g = j_cur + q;
               h = k_cur + r;
               if ((f<1)|(f>ncellx)|(g<1)|(g>ncelly)|(h<1)|(h>ncellz))
                  continue;
               else
                  if (mask(f,g,h) > 0) %push nonzero neighbors onto the stack
                     mask(f,g,h) = 0;
                     fx = cell_x(f);
                     gy = cell_y(g);
                     hz = cell_z(h);
                     [new_stack_top,stack] = push(f,g,h,fx,gy,hz,cur_stack_top,stack);
                     cur_stack_top = new_stack_top;
                  end %end if
               end %end if else
            end %end if
         end %end for r
      end %end for q
   end %end for p
```

```
      end %end while
   end %end if
end %end for nz_vox
```

A.1.1.6 OBJECT_TEST.M

```
function [true_obj_no, true_object]=Object_test(obj_no,object)
%OBJECT_TEST - True objects in a scanned scene are assumed to have
%            three dimensions. This function extracts from the
%            array of potential objects those that have a three
%            dimensional structure of voxels.
%
%%Object is an array of sructures defined as follow:
%object(obj_no).size  is the number of voxels in the object = count
%object(obj_no).index(count,1) = ii; i index for current voxel
%object(obj_no).index(count,2) = jj; j index for current voxel
%object(obj_no).index(count,3) = kk; k index for current voxel
%object(obj_no).coord(count,1) = cell_x(ii); center x for current voxel
%object(obj_no).coord(count,2) = cell_y(jj); center y for current voxel
%object(obj_n0).coord(count,3) = cell_z(kk); ccenter z for current voxel
%
%obj_no is the number of potential objects
%
%*********************************************************************
%
%
% In this function we will use the object indices as surrogate coordinates
% The algorithm will proceed as follows:
% For each object we test object(obj).size. If object(obj).size = 1,2,or 3
% we proceed to the next object in the list since these could be at most
% a point, a line, and a plane. For an object with 4 points we generate the
% matrix test_mat = [i1 j1 k1 1;i2 j2 k2 1;i3 j3 k3 1;i4 j4 k4 1] and
% compute the condition number cond(test_mat). If cond(test_mat) < 1/eps
% then the points form a three dimensional object. If not the points
% can form at most a plane. If there are more than 4 points then we proceed
% as follows: We begin building the test_mat by setting i1 j1 k1 as the
% first triple ii jj kk for the object, then set i2 j2 k2 as the second
% triple for the object. This produces a line. In order to produce a plane
% we need to find a non-collinear point. This can be done by examining inner
% products. Once a non-collinear point is found we have i3 j3 k3 and we
% test the condition of test_mat = [i1 j1 k1 1;i2 j2 k2 1;i3 j3 k3 1;ii jj
% kk 1] for each ii jj kk triple after i3 j3 k3 is found. If the condition
% test passes then this object is three dimensional and can be identified
% as a true object.
%
% This function also includes a fall through in case there are no identifiable three
% dimensional objects
%
%************************************************************************
%
true_obj_no = 0;
for obj = 1:obj_no
   object_size = object(obj).size;
   if ( object_size > 3)
```

```
if (object_size == 4)
   for r = 1:4
      ii(r) = object(obj).index(r,1);
      jj(r) = object(obj).index(r,2);
      kk(r) = object(obj).index(r,3);
   end %for r = 1:4
   test_mat = [ii(1) jj(1) kk(1) 1;ii(2) jj(2) kk(2) 1;...
         ii(3) jj(3) kk(3) 1;ii(4) jj(4) kk(4) 1];
   if (cond(test_mat) < (1/eps))
      true_obj_no = true_obj_no + 1;
      true_object(true_obj_no).size = 4;
         for rr = 1:4
            true_object(true_obj_no).index(rr,1) = object(obj).index(rr,1);
            true_object(true_obj_no).index(rr,2) = object(obj).index(rr,2);
            true_object(true_obj_no).index(rr,3) = object(obj).index(rr,3);
            true_object(true_obj_no).coord(rr,1) = object(obj).coord(rr,1);
            true_object(true_obj_no).coord(rr,2) = object(obj).coord(rr,2);
            true_object(true_obj_no).coord(rr,3) = object(obj).coord(rr,3);
         end %for rr = 1:4
   end %if (cond(test_mat) < (1/eps))
else %object size > 4
   for r = 1:2
      ii(r) = object(obj).index(r,1);
      jj(r) = object(obj).index(r,2);
      kk(r) = object(obj).index(r,3);
   end %for r = 1:2
   % Form the unit vector from point 1 to 2
   v = [ii(2)-ii(1);jj(2)-jj(1);kk(2)-kk(1)];
   v = v/norm(v);
   % Find a non-collinear point and test other points for
   % non-coplanarity
   found3 = 0;
   found4 = 0;
   for r = 3:object_size
      if (((found3 == 0)&(found4 == 0)) | ((found3 == 1)&(found4 == 0)))
         ii(r) = object(obj).index(r,1);
         jj(r) = object(obj).index(r,2);
         kk(r) = object(obj).index(r,3);
         if (found3 == 0)
            % Form a unit vector between point r and 1
            p = [ii(r)-ii(1);jj(r)-jj(1);kk(r)-kk(1)];
            p = p/norm(p);
            % Form the inner product and test whether its absolute
            % value is near 1, i.e collinear
            v_dot_p = v(1)*p(1) + v(2)*p(2) + v(3)*p(3);
            if (abs(abs(v_dot_p)-1) > 10^(-8))
               ii(3) = object(obj).index(r,1);
               jj(3) = object(obj).index(r,2);
               kk(3) = object(obj).index(r,3);
               found3 = 1;
            end %if (abs(abs(v_dot_p)-1) > 10^(-8))
            if (found3 == 1)
               continue;
            end %if (found3 == 1)
         else
            % Form test matrix and check condition
```

```
            test_mat = [ii(1) jj(1) kk(1) 1; ii(2) jj(2) kk(2) 1;...
                ii(3) jj(3) kk(3) 1; ii(r) jj(r) kk(r) 1];
            if (cond(test_mat) < (1/eps))
              % Increment the true object number and add to
              % the new structure
              true_obj_no = true_obj_no + 1;
              true_object(true_obj_no).size = object(obj).size;
              for rr = 1:true_object(true_obj_no).size
                true_object(true_obj_no).index(rr,1) = object(obj).index(rr,1);
                true_object(true_obj_no).index(rr,2) = object(obj).index(rr,2);
                true_object(true_obj_no).index(rr,3) = object(obj).index(rr,3);
                true_object(true_obj_no).coord(rr,1) = object(obj).coord(rr,1);
                true_object(true_obj_no).coord(rr,2) = object(obj).coord(rr,2);
                true_object(true_obj_no).coord(rr,3) = object(obj).coord(rr,3);
              end %for rr = 1:true_object(true_obj_no).size
                found4 = 1;
            end %if (cond(test_mat) < (1/eps))
          end %if (found3 == 0)
        elseif ((found3 == 1)&(found4 == 1))
          break;
        end %if (((found3 == 0)&(found4 == 0)) | ((found3 == 1)&(found4 == 0)))
      end %for r = 3:object_size
    end %if (object_size == 4)
  end %if ( object_size > 3)
end %for obj = 1:obj_no
if (true_obj_no == 0) % if there are no identifiable 3D objects then pass back the input params
  true_obj_no = obj_no;
  for obj = 1:obj_no
  true_object(obj).size = object(obj).size;
    for rr = 1:true_object(obj).size
      true_object(obj).index(rr,1) = object(obj).index(rr,1);
      true_object(obj).index(rr,2) = object(obj).index(rr,2);
      true_object(obj).index(rr,3) = object(obj).index(rr,3);
      true_object(obj).coord(rr,1) = object(obj).coord(rr,1);
      true_object(obj).coord(rr,2) = object(obj).coord(rr,2);
      true_object(obj).coord(rr,3) = object(obj).coord(rr,3);
    end %for rr = 1:true_object(obj).size
  end
end
```

A.1.1.7 PUSH.M

```
function [new_stack_top, stack] = push(ii,jj,kk,xx,yy,zz,cur_stack_top,stack)
%PUSH - pushes voxel identification onto the stack
m = cur_stack_top;
m = m+1;
stack(m).i(1,1) = ii;
stack(m).j(1,1) = jj;
stack(m).k(1,1) = kk;
stack(m).x(1,1) = xx;
stack(m).y(1,1) = yy;
stack(m).z(1,1) = zz;
new_stack_top = m;
```

A.1.1.8 POP.M

```
function new_stack_top = pop(cur_stack_top)
%POP - pop an element off the stack
new_stack_top = cur_stack_top - 1;
```

A.1.2 Object Identification by TINs Scripts

The script included here assumes that there is an input file of data points associated with predefined potential objects. The data points associated with potential objects are separated in the file by a row of 9's. The current script is essentially a copy of the main script for binning segmentation but eliminates the portion of that script involved with potential object segmentation.

```
clear all;
%Open file for Excel comma separated output
fid3 = fopen('All_seg_output.txt','w');
% set up header lines
fprintf(fid3,'File, I-Beam, 8 Point, Location, of, CM, Pose from, Site,,
Metrology, System, ,Bounding, Box, , ,Error\n');
fprintf(fid3,'Name, Selected, Error, x,y,z, neg x axis
(deg),Point,x,y,z,Point,x,y,z,dx,dy,dz\n');
%*****************************************************************
% SMS to scanner coordinate transformation based on reference sphere
% centers
% Transformation is given by:
% Scanner_coord = Transform*(sms_coord - cg) + cf
% Transformation is not restricted to rotation
%*****************************************************************
cg = [10.13450; -2.051750; -0.3965000];
cf = [-2.655250; 4.259250; -0.2567500];
Transform = [0.4362677 0.8998131 0.0026162; -0.8998142 0.4362716 -0.0011596;
-0.0021848 -0.0018482 0.9999958];
%*****************************************************************
% Read in the database of I-beams and generate points
%*****************************************************************
beam_database = 'I_beam_database.txt';
%Database values defined in inches
fid_db = fopen(beam_database,'r');
no_beams = fscanf(fid_db,'%d',1);
cv_ib = zeros(3,3);
%Create points that define the I-beams with the zero located at the center
%of mass, convert to meters by multiplying by 0.0254 in/m
for i = 1:no_beams
    I_beam(i,1).name = fscanf(fid_db,'%s',1);
    I_beam(i,1).length = fscanf(fid_db,'%f',1);
    I_beam(i,1).depth = fscanf(fid_db,'%f',1);
    I_beam(i,1).flange_width = fscanf(fid_db,'%f',1);
    I_beam(i,1).flange_thick = fscanf(fid_db,'%f',1);
    I_beam(i,1).web_thick = fscanf(fid_db,'%f',1);
    I_beam(i,1).web_height = I_beam(i,1).depth - 2*I_beam(i,1).flange_thick;
    I_beam(i,1).x(1,1) = 0.5*I_beam(i,1).flange_width;
```

```
I_beam(i,1).y(1,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(1,1) = -0.5*I_beam(i,1).depth;
I_beam(i,1).x(2,1) = -0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(2,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(2,1) = -0.5*I_beam(i,1).depth;
I_beam(i,1).x(3,1) = 0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(3,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(3,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(4,1) = -0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(4,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(4,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(5,1) = 0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(5,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(5,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(6,1) = -0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(6,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(6,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(7,1) = 0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(7,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(7,1) = 0.5*I_beam(i,1).web_height;
I_beam(i,1).x(8,1) = -0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(8,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(8,1) = 0.5*I_beam(i,1).web_height;
I_beam(i,1).x(9,1) = 0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(9,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(9,1) = 0.5*I_beam(i,1).web_height;
I_beam(i,1).x(10,1) = -0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(10,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(10,1) = 0.5*I_beam(i,1).web_height;
I_beam(i,1).x(11,1) = 0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(11,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(11,1) = 0.5*I_beam(i,1).depth;
I_beam(i,1).x(12,1) = -0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(12,1) = 0.5*I_beam(i,1).length;
I_beam(i,1).z(12,1) = 0.5*I_beam(i,1).depth;
I_beam(i,1).x(13,1) = 0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(13,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(13,1) = -0.5*I_beam(i,1).depth;
I_beam(i,1).x(14,1) = -0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(14,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(14,1) = -0.5*I_beam(i,1).depth;
I_beam(i,1).x(15,1) = 0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(15,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(15,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(16,1) = -0.5*I_beam(i,1).flange_width;
I_beam(i,1).y(16,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(16,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(17,1) = 0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(17,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(17,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(18,1) = -0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(18,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(18,1) = -0.5*I_beam(i,1).web_height;
I_beam(i,1).x(19,1) = 0.5*I_beam(i,1).web_thick;
I_beam(i,1).y(19,1) = -0.5*I_beam(i,1).length;
I_beam(i,1).z(19,1) = 0.5*I_beam(i,1).web_height;
I_beam(i,1).x(20,1) = -0.5*I_beam(i,1).web_thick;
```

```
    I_beam(i,1).y(20,1) = -0.5*I_beam(i,1).length;
    I_beam(i,1).z(20,1) = 0.5*I_beam(i,1).web_height;
    I_beam(i,1).x(21,1) = 0.5*I_beam(i,1).flange_width;
    I_beam(i,1).y(21,1) = -0.5*I_beam(i,1).length;
    I_beam(i,1).z(21,1) = 0.5*I_beam(i,1).web_height;
    I_beam(i,1).x(22,1) = -0.5*I_beam(i,1).flange_width;
    I_beam(i,1).y(22,1) = -0.5*I_beam(i,1).length;
    I_beam(i,1).z(22,1) = 0.5*I_beam(i,1).web_height;
    I_beam(i,1).x(23,1) = 0.5*I_beam(i,1).flange_width;
    I_beam(i,1).y(23,1) = -0.5*I_beam(i,1).length;
    I_beam(i,1).z(23,1) = 0.5*I_beam(i,1).depth;
    I_beam(i,1).x(24,1) = -0.5*I_beam(i,1).flange_width;
    I_beam(i,1).y(24,1) = -0.5*I_beam(i,1).length;
    I_beam(i,1).z(24,1) = 0.5*I_beam(i,1).depth;
    %convert to meters
    I_beam(i,1).x(:,1) = I_beam(i,1).x(:,1)*0.0254;
    I_beam(i,1).y(:,1) = I_beam(i,1).y(:,1)*0.0254;
    I_beam(i,1).z(:,1) = I_beam(i,1).z(:,1)*0.0254;
    nn_ib = 24;
    xx_ib = I_beam(i,1).x;
    yy_ib = I_beam(i,1).y;
    zz_ib = I_beam(i,1).z;
    [cm_ib,pr_ax_ib,cv_ib,eigvec_ib,eigval_ib] =
princ_axes(nn_ib,xx_ib,yy_ib,zz_ib); %get center of mass and principle axes
    I_beam(i,1).ctr_mass.x = cm_ib(1,1);
    I_beam(i,1).ctr_mass.y = cm_ib(2,1);
    I_beam(i,1).ctr_mass.z = cm_ib(3,1);
    I_beam(i,1).R.x = pr_ax_ib(1,1);
    I_beam(i,1).R.y = pr_ax_ib(2,1);
    I_beam(i,1).R.z = pr_ax_ib(3,1);
    I_beam(i,1).S.x = pr_ax_ib(1,2);
    I_beam(i,1).S.y = pr_ax_ib(2,2);
    I_beam(i,1).S.z = pr_ax_ib(3,2);
    I_beam(i,1).T.x = pr_ax_ib(1,3);
    I_beam(i,1).T.y = pr_ax_ib(2,3);
    I_beam(i,1).T.z = pr_ax_ib(3,3);
    for np = 1:nn_ib %compute inner products of points and principle axes
        R_dot_P_I(np,1) = I_beam(i,1).R.x * xx_ib(np,1) + I_beam(i,1).R.y *
yy_ib(np,1)...
            + I_beam(i,1).R.z * zz_ib(np,1);
        S_dot_P_I(np,1) = I_beam(i,1).S.x * xx_ib(np,1) + I_beam(i,1).S.y *
yy_ib(np,1)...
            + I_beam(i,1).S.z * zz_ib(np,1);
        T_dot_P_I(np,1) = I_beam(i,1).T.x * xx_ib(np,1) + I_beam(i,1).T.y *
yy_ib(np,1)...
            + I_beam(i,1).T.z * zz_ib(np,1);
    end
    max_R_dot_P_I = max(R_dot_P_I); min_R_dot_P_I = min(R_dot_P_I);
    max_S_dot_P_I = max(S_dot_P_I); min_S_dot_P_I = min(S_dot_P_I);
    max_T_dot_P_I = max(T_dot_P_I); min_T_dot_P_I = min(T_dot_P_I);
    aa = ( max_R_dot_P_I - min_R_dot_P_I)/2;
    bb = ( max_S_dot_P_I - min_S_dot_P_I)/2;
    cc = ( max_T_dot_P_I - min_T_dot_P_I)/2;
    % Compute the bounding box center
    I_beam(i,1).Q.x = cm_ib(1,1);
    I_beam(i,1).Q.y = cm_ib(2,1);
    I_beam(i,1).Q.z = cm_ib(3,1);
```

```
% Get the eight corners of the box.
xt = I_beam(i,1).Q.x + aa*I_beam(i,1).R.x...
    + bb*I_beam(i,1).S.x + cc*I_beam(i,1).T.x;
I_beam(i,1).corner(1,1).x1 =xt;
yt = I_beam(i,1).Q.y + aa*I_beam(i,1).R.y...
    + bb*I_beam(i,1).S.y + cc*I_beam(i,1).T.y;
I_beam(i,1).corner(1,1).y1 = yt;
zt = I_beam(i,1).Q.z + aa*I_beam(i,1).R.z...
    + bb*I_beam(i,1).S.z + cc*I_beam(i,1).T.z;
I_beam(i,1).corner(1,1).z1 = zt;
xt = I_beam(i,1).Q.x + aa*I_beam(i,1).R.x...
    + bb*I_beam(i,1).S.x - cc*I_beam(i,1).T.x;
I_beam(i,1).corner(2,1).x1 = xt;
yt = I_beam(i,1).Q.y + aa*I_beam(i,1).R.y...
    + bb*I_beam(i,1).S.y - cc*I_beam(i,1).T.y;
I_beam(i,1).corner(2,1).y1 = yt;
zt = I_beam(i,1).Q.z + aa*I_beam(i,1).R.z...
    + bb*I_beam(i,1).S.z - cc*I_beam(i,1).T.z;
I_beam(i,1).corner(2,1).z1 = zt;
xt = I_beam(i,1).Q.x + aa*I_beam(i,1).R.x...
    - bb*I_beam(i,1).S.x + cc*I_beam(i,1).T.x;
I_beam(i,1).corner(3,1).x1 = xt;
yt = I_beam(i,1).Q.y + aa*I_beam(i,1).R.y...
    - bb*I_beam(i,1).S.y + cc*I_beam(i,1).T.y;
I_beam(i,1).corner(3,1).y1 = yt;
zt = I_beam(i,1).Q.z + aa*I_beam(i,1).R.z...
    - bb*I_beam(i,1).S.z + cc*I_beam(i,1).T.z;
I_beam(i,1).corner(3,1).z1 = zt;
xt = I_beam(i,1).Q.x + aa*I_beam(i,1).R.x...
    - bb*I_beam(i,1).S.x - cc*I_beam(i,1).T.x;
I_beam(i,1).corner(4,1).x1 = xt;
yt = I_beam(i,1).Q.y + aa*I_beam(i,1).R.y...
    - bb*I_beam(i,1).S.y - cc*I_beam(i,1).T.y;
I_beam(i,1).corner(4,1).y1 = yt;
zt = I_beam(i,1).Q.z + aa*I_beam(i,1).R.z...
    - bb*I_beam(i,1).S.z - cc*I_beam(i,1).T.z;
I_beam(i,1).corner(4,1).z1 = zt;
xt = I_beam(i,1).Q.x - aa*I_beam(i,1).R.x...
    + bb*I_beam(i,1).S.x + cc*I_beam(i,1).T.x;
I_beam(i,1).corner(5,1).x1 = xt;
yt = I_beam(i,1).Q.y - aa*I_beam(i,1).R.y...
    + bb*I_beam(i,1).S.y + cc*I_beam(i,1).T.y;
I_beam(i,1).corner(5,1).y1 = yt;
zt = I_beam(i,1).Q.z - aa*I_beam(i,1).R.z...
    + bb*I_beam(i,1).S.z + cc*I_beam(i,1).T.z;
I_beam(i,1).corner(5,1).z1 = zt;
xt = I_beam(i,1).Q.x - aa*I_beam(i,1).R.x...
    + bb*I_beam(i,1).S.x - cc*I_beam(i,1).T.x;
I_beam(i,1).corner(6,1).x1 = xt;
yt = I_beam(i,1).Q.y - aa*I_beam(i,1).R.y...
    + bb*I_beam(i,1).S.y - cc*I_beam(i,1).T.y;
I_beam(i,1).corner(6,1).y1 = yt;
zt = I_beam(i,1).Q.z - aa*I_beam(i,1).R.z...
    + bb*I_beam(i,1).S.z - cc*I_beam(i,1).T.z;
I_beam(i,1).corner(6,1).z1 = zt;
xt = I_beam(i,1).Q.x - aa*I_beam(i,1).R.x...
    - bb*I_beam(i,1).S.x + cc*I_beam(i,1).T.x;
```

```
    I_beam(i,1).corner(7,1).x1 = xt;
    yt = I_beam(i,1).Q.y - aa*I_beam(i,1).R.y...
        - bb*I_beam(i,1).S.y + cc*I_beam(i,1).T.y;
    I_beam(i,1).corner(7,1).y1 = yt;
    zt = I_beam(i,1).Q.z - aa*I_beam(i,1).R.z...
        - bb*I_beam(i,1).S.z + cc*I_beam(i,1).T.z;
    I_beam(i,1).corner(7,1).z1 = zt;
    xt = I_beam(i,1).Q.x - aa*I_beam(i,1).R.x...
        - bb*I_beam(i,1).S.x - cc*I_beam(i,1).T.x;
    I_beam(i,1).corner(8,1).x1 = xt;
    yt = I_beam(i,1).Q.y - aa*I_beam(i,1).R.y...
        - bb*I_beam(i,1).S.y - cc*I_beam(i,1).T.y;
    I_beam(i,1).corner(8,1).y1 = yt;
    zt = I_beam(i,1).Q.z - aa*I_beam(i,1).R.z...
        - bb*I_beam(i,1).S.z - cc*I_beam(i,1).T.z;
    I_beam(i,1).corner(8,1).z1 = zt;
end
fclose(fid_db);
%******************************************************************
% Load the SMS Flange Corner Point file
%******************************************************************
sms_file = 'SMS_I_beam_points.txt';
sms = load(sms_file);
[sm,sn] = size(sms);
beam_no = zeros(sm,1);
corner_no = zeros(sm,1);
rot_angle = zeros(sm,1);
sms_x = zeros(sm,1);
sms_y = zeros(sm,1);
sms_z = zeros(sm,1);
beam_no(:,1) = sms(:,1);
corner_no(:,1) = sms(:,2);
rot_angle(:,1) = sms(:,3);
sms_x(:,1) = sms(:,4);
sms_y(:,1) = sms(:,5);
sms_z(:,1) = sms(:,6);
%**************************************************************
%Load the segment file list and process each file
%**************************************************************
fid1 = fopen('All_seg_files.txt','r');
while feof(fid1)==0 %see end for end statenebt
    %clear variables
    clear ladar_file max_x min_x max_y min_y max_z min_z ncellx ncelly ncellz
    clear del_cellx del_celly del_cellz cell_x dx cell_y dy cell_z dz
bin_count
    clear nz_i_bins nz_j_binx nz_k_bins count lnz nz_i_bins_x nz_j_bins_y
nz_k_bins_z
    clear Max_string_length Max_L nz_i_bins_new nz_j_bins_new nz_k_bins_new
    clear nz_i_bins_x_new nz_j_bins_y_new nz_k_bins_z_new potential_object
obj_no
    clear n_obj object R_dot_P S_dot_P T_dot_P max_R_dot_P min_R_dot_P
max_S_dot_P
    clear min_S_dot_P max_T_dot_P min_T_dot_P aa bb cc cc1 AA rt Rot T H mpt
trmpt
    clear comparison_error min_error i_min obj_min
ladar_file = fgetl(fid1);
disp(sprintf('Processing file = %s\n',ladar_file));
```

```
a = load(ladar_file);
[m,n] = size(a);
xtemp = zeros(m,1);
ytemp = zeros(m,1);
ztemp = zeros(m,1);
xtemp(:,1) = a(:,1);
ytemp(:,1) = a(:,2);
ztemp(:,1) = a(:,3);
x = xtemp;
y = ytemp;
z = ztemp;
separators = find(xtemp > 9e6);
n_obj = length(separators);
cv_obj = zeros(3,3);
for obj = 1:n_obj
    if (obj == 1)
        nn = separators(1) - 1;
        coord = zeros(nn,3);
        x_coord = zeros(nn,1);
        y_coord = zeros(nn,1);
        z_coord = zeros(nn,1);
        object(obj,1).size = nn;
        object(obj,1).coord(1:nn,1) = xtemp(1:nn,1);
        object(obj,1).coord(1:nn,2) = ytemp(1:nn,1);
        object(obj,1).coord(1:nn,3) = ztemp(1:nn,1);
    else
        nn = separators(obj)-separators(obj-1)-1;
        n1 = separators(obj-1);
        coord = zeros(nn,3);
        x_coord = zeros(nn,1);
        y_coord = zeros(nn,1);
        z_coord = zeros(nn,1);
        object(obj,1).size = nn;
        object(obj,1).coord(1:nn,1) = xtemp(n1+1:n1+nn,1);
        object(obj,1).coord(1:nn,2) = ytemp(n1+1:n1+nn,1);
        object(obj,1).coord(1:nn,3) = ztemp(n1+1:n1+nn,1);
    end
    coord = zeros(nn,3);
    coord = object(obj,1).coord;
    x_coord(1:nn,1) = coord(1:nn,1);
    y_coord(1:nn,1) = coord(1:nn,2);
    z_coord(1:nn,1) = coord(1:nn,3);
    [cm,pr_ax,cv_obj,eigvec_obj,eigval_obj] =
princ_axes(nn,x_coord,y_coord,z_coord); %get center of mass and principle
axes
    object(obj,1).ctr_mass.x = cm(1,1);
    object(obj,1).ctr_mass.y = cm(2,1);
    object(obj,1).ctr_mass.z = cm(3,1);
    object(obj,1).R.x = pr_ax(1,1);
    object(obj,1).R.y = pr_ax(2,1);
    object(obj,1).R.z = pr_ax(3,1);
    object(obj,1).S.x = pr_ax(1,2);
    object(obj,1).S.y = pr_ax(2,2);
    object(obj,1).S.z = pr_ax(3,2);
    object(obj,1).T.x = pr_ax(1,3);
    object(obj,1).T.y = pr_ax(2,3);
    object(obj,1).T.z = pr_ax(3,3);
```

```
    if (det(eigvec_obj) < 0)
        object(obj,1).R.x = -object(obj,1).R.x;
        object(obj,1).R.y = -object(obj,1).R.y;
        object(obj,1).R.z = -object(obj,1).R.z;
    end
    R_dot_P = zeros(nn,1);
    S_dot_P = zeros(nn,1);
    T_dot_P = zeros(nn,1);
     %compute inner products of points and principle axes
        R_dot_P(:,1) = object(obj,1).R.x * x_coord(:,1) + object(obj,1).R.y *
y_coord(:,1)...
            + object(obj,1).R.z * z_coord(:,1);
        S_dot_P(:,1) = object(obj,1).S.x * x_coord(:,1) + object(obj,1).S.y *
y_coord(:,1)...
            + object(obj,1).S.z * z_coord(:,1);
        T_dot_P(:,1) = object(obj,1).T.x * x_coord(:,1) + object(obj,1).T.y *
y_coord(:,1)...
            + object(obj,1).T.z * z_coord(:,1);
    max_R_dot_P(obj,1) = max(R_dot_P); min_R_dot_P(obj,1) = min(R_dot_P);
    max_S_dot_P(obj,1) = max(S_dot_P); min_S_dot_P(obj,1) = min(S_dot_P);
    max_T_dot_P(obj,1) = max(T_dot_P); min_T_dot_P(obj,1) = min(T_dot_P);
    a(obj,1) = ( max_R_dot_P(obj,1) - min_R_dot_P(obj,1))/2;
    b(obj,1) = ( max_S_dot_P(obj,1) - min_S_dot_P(obj,1))/2;
    c(obj,1) = ( max_T_dot_P(obj,1) - min_T_dot_P(obj,1))/2;
    aa = a(obj,1);
    bb = b(obj,1);
    cc = c(obj,1);
    % Compute the bounding box center
    object(obj,1).Q.x = cm(1,1);
    object(obj,1).Q.y = cm(2,1);
    object(obj,1).Q.z = cm(3,1);
    % Get the eight corners of the box.
    object(obj,1).corner(1,1).x = object(obj,1).Q.x + aa*object(obj,1).R.x...
        + bb*object(obj,1).S.x + cc*object(obj,1).T.x;
    object(obj,1).corner(1,1).y = object(obj,1).Q.y + aa*object(obj,1).R.y...
        + bb*object(obj,1).S.y + cc*object(obj,1).T.y;
    object(obj,1).corner(1,1).z = object(obj,1).Q.z + aa*object(obj,1).R.z...
        + bb*object(obj,1).S.z + cc*object(obj,1).T.z;
    object(obj,1).corner(2,1).x = object(obj,1).Q.x + aa*object(obj,1).R.x...
        + bb*object(obj,1).S.x - cc*object(obj,1).T.x;
    object(obj,1).corner(2,1).y = object(obj,1).Q.y + aa*object(obj,1).R.y...
        + bb*object(obj,1).S.y - cc*object(obj,1).T.y;
    object(obj,1).corner(2,1).z = object(obj,1).Q.z + aa*object(obj,1).R.z...
        + bb*object(obj,1).S.z - cc*object(obj,1).T.z;
    object(obj,1).corner(3,1).x = object(obj,1).Q.x + aa*object(obj,1).R.x...
        - bb*object(obj,1).S.x + cc*object(obj,1).T.x;
    object(obj,1).corner(3,1).y = object(obj,1).Q.y + aa*object(obj,1).R.y...
        - bb*object(obj,1).S.y + cc*object(obj,1).T.y;
    object(obj,1).corner(3,1).z = object(obj,1).Q.z + aa*object(obj,1).R.z...
        - bb*object(obj,1).S.z + cc*object(obj,1).T.z;
    object(obj,1).corner(4,1).x = object(obj,1).Q.x + aa*object(obj,1).R.x...
        - bb*object(obj,1).S.x - cc*object(obj,1).T.x;
    object(obj,1).corner(4,1).y = object(obj,1).Q.y + aa*object(obj,1).R.y...
        - bb*object(obj,1).S.y - cc*object(obj,1).T.y;
    object(obj,1).corner(4,1).z = object(obj,1).Q.z + aa*object(obj,1).R.z...
        - bb*object(obj,1).S.z - cc*object(obj,1).T.z;
    object(obj,1).corner(5,1).x = object(obj,1).Q.x - aa*object(obj,1).R.x...
```

```
        + bb*object(obj,1).S.x + cc*object(obj,1).T.x;
    object(obj,1).corner(5,1).y = object(obj,1).Q.y - aa*object(obj,1).R.y...
        + bb*object(obj,1).S.y + cc*object(obj,1).T.y;
    object(obj,1).corner(5,1).z = object(obj,1).Q.z - aa*object(obj,1).R.z...
        + bb*object(obj,1).S.z + cc*object(obj,1).T.z;
    object(obj,1).corner(6,1).x = object(obj,1).Q.x - aa*object(obj,1).R.x...
        + bb*object(obj,1).S.x - cc*object(obj,1).T.x;
    object(obj,1).corner(6,1).y = object(obj,1).Q.y - aa*object(obj,1).R.y...
        + bb*object(obj,1).S.y - cc*object(obj,1).T.y;
    object(obj,1).corner(6,1).z = object(obj,1).Q.z - aa*object(obj,1).R.z...
        + bb*object(obj,1).S.z - cc*object(obj,1).T.z;
    object(obj,1).corner(7,1).x = object(obj,1).Q.x - aa*object(obj,1).R.x...
        - bb*object(obj,1).S.x + cc*object(obj,1).T.x;
    object(obj,1).corner(7,1).y = object(obj,1).Q.y - aa*object(obj,1).R.y...
        - bb*object(obj,1).S.y + cc*object(obj,1).T.y;
    object(obj,1).corner(7,1).z = object(obj,1).Q.z - aa*object(obj,1).R.z...
        - bb*object(obj,1).S.z + cc*object(obj,1).T.z;
    object(obj,1).corner(8,1).x = object(obj,1).Q.x - aa*object(obj,1).R.x...
        - bb*object(obj,1).S.x - cc*object(obj,1).T.x;
    object(obj,1).corner(8,1).y = object(obj,1).Q.y - aa*object(obj,1).R.y...
        - bb*object(obj,1).S.y - cc*object(obj,1).T.y;
    object(obj,1).corner(8,1).z = object(obj,1).Q.z - aa*object(obj,1).R.z...
        - bb*object(obj,1).S.z - cc*object(obj,1).T.z;
end
for i = 1:no_beams
    for obj = 1:n_obj
        %Next find the rotation that aligns the principal axes
        %of the model I_beam box with the object box
            cc1 = zeros(9,1);
            %set up the right hand side of the object eigenvectors
            cc1(1,1) = object(obj,1).R.x;
            cc1(2,1) = object(obj,1).R.y;
            cc1(3,1) = object(obj,1).R.z;
            cc1(4,1) = object(obj,1).S.x;
            cc1(5,1) = object(obj,1).S.y;
            cc1(6,1) = object(obj,1).S.z;
            cc1(7,1) = object(obj,1).T.x;
            cc1(8,1) = object(obj,1).T.y;
            cc1(9,1) = object(obj,1).T.z;
            %set up the 9 x 9 matrix of model eigenvectors
            AA = zeros(9,9);
            AA(1,1) = I_beam(i).R.x;
            AA(1,2) = I_beam(i).R.y;
            AA(1,3) = I_beam(i).R.z;
            AA(2,4) = I_beam(i).R.x;
            AA(2,5) = I_beam(i).R.y;
            AA(2,6) = I_beam(i).R.z;
            AA(3,7) = I_beam(i).R.x;
            AA(3,8) = I_beam(i).R.y;
            AA(3,9) = I_beam(i).R.z;
            AA(4,1) = I_beam(i).S.x;
            AA(4,2) = I_beam(i).S.y;
            AA(4,3) = I_beam(i).S.z;
            AA(5,4) = I_beam(i).S.x;
            AA(5,5) = I_beam(i).S.y;
            AA(5,6) = I_beam(i).S.z;
            AA(6,7) = I_beam(i).S.x;
```

```
            AA(6,8) = I_beam(i).S.y;
            AA(6,9) = I_beam(i).S.z;
            AA(7,1) = I_beam(i).T.x;
            AA(7,2) = I_beam(i).T.y;
            AA(7,3) = I_beam(i).T.z;
            AA(8,4) = I_beam(i).T.x;
            AA(8,5) = I_beam(i).T.y;
            AA(8,6) = I_beam(i).T.z;
            AA(9,7) = I_beam(i).T.x;
            AA(9,8) = I_beam(i).T.y;
            AA(9,9) = I_beam(i).T.z;
            %solve least squares for the rotation matrix
            rt = zeros(9,1);
            Rot = zeros(3,3);
            rt = AA\cc1;
            %Make the rotation matrix a 2D array
            for ii = 1:3
                for jj = 1:3
                    kk = (ii-1)*3 + jj;
                    Rot(ii,jj) = rt(kk,1);
                end
            end
            %Apply the translation and rotation to the model data with a
homogeneous
            %transformation
            T(1,1) = object(obj,1).ctr_mass.x;
            T(2,1) = object(obj,1).ctr_mass.y;
            T(3,1) = object(obj,1).ctr_mass.z;
            %Set up homogeneous transformation
            H = zeros(4,4);
            for ii = 1:3
                for jj = 1:3
                    H(ii,jj) = Rot(ii,jj);
                end
            end
            H(1,4) = T(1,1);
            H(2,4) = T(2,1);
            H(3,4) = T(3,1);
            H(4,4) = 1.0;
            %Rotate and translate the bounding box for each beam model
            %Compute the sum of squares differences between the true object
            %and the beam
            error = 0;
            for v = 1:8
                mpt(1,1) = I_beam(i).corner(v).x1;
                mpt(2,1) = I_beam(i).corner(v).y1;
                mpt(3,1) = I_beam(i).corner(v).z1;
                mpt(4,1) = 1.0;
                trmpt = H*mpt; %apply homogeneous transformation
                I_beam(i).corner(v).trrot(obj,1).x2 = trmpt(1,1);
                I_beam(i).corner(v).trrot(obj,1).y2 = trmpt(2,1);
                I_beam(i).corner(v).trrot(obj,1).z2 = trmpt(3,1);
                error = error + (object(obj,1).corner(v).x -...
                    I_beam(i).corner(v).trrot(obj,1).x2)^2 +
(object(obj,1).corner(v).y -...
                    I_beam(i).corner(v).trrot(obj,1).y2)^2 +
(object(obj,1).corner(v).z -...
```

```
                    I_beam(i).corner(v).trrot(obj,1).z2)^2;
            end
            comparison_error(i,obj) = sqrt(error);
    end
end
min_error = 1e+10;
for i = 1:no_beams
    for obj = 1:n_obj
        if (comparison_error(i,obj) < min_error)
            min_error = comparison_error(i,obj);
            i_min = i;
            obj_min = obj;
        end
    end
end
pose_angle = zeros(n_obj,1);
pose_angle(obj_min) = (180/pi)*
atan(object(obj_min).R.y/object(obj_min).R.x);
if (pose_angle(obj_min) < 0)
    pose_angle(obj_min) = 180 + pose_angle(obj_min);
end
%***************************************************************************
% Reading the sms measurements for the four top flange corners
% sms point a = bounding box corner 5
% sms point b = bounding box corner 6
% sms point c = bounding box corner 1
% sms point d = bounding box corner 2
%***************************************************************************
% Getting table entry for sms point a (bounding box corner 5)
if((ladar_file(1) =='A')& (ladar_file(2) == '0'))
    entry = 1;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '3') & (ladar_file(3) ==
'0'))
    entry = 5;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '4') & (ladar_file(3) ==
'5'))
     entry = 9;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '6') & (ladar_file(3) ==
'0'))

     entry = 13;
elseif((ladar_file(1) =='A') & (ladar_file(2) == '9') & (ladar_file(3) ==
'0'))
     entry = 17;
elseif((ladar_file(1) =='B')& (ladar_file(2) == '0'))
     entry = 21;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '3') & (ladar_file(3) ==
'0'))
     entry = 25;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '4') & (ladar_file(3) ==
'5'))
     entry = 29;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '6') & (ladar_file(3) ==
'0'))
     entry = 33;
elseif((ladar_file(1) =='B') & (ladar_file(2) == '9') & (ladar_file(3) ==
'0'))
```

```
     entry = 37;
end
% Put sms points into scanner frame for comparison. File name provides the
% entry point into the sms corner measurement file
scanner_pt = zeros(3,1);
sms_pt = zeros(3,1);
%At this point we need to determine which points to compare
% The four sms points are defined as a, b, c, and d. The four top bounding
% box points are 1,2, 5,and 6. We either (1) compare a to 1, b to 2, c to 5,
% and d to 6 or (2) we compare a to 5, b to 6, c to 1, and d to 2
% To determine which set of comparisons to make we will generate two unit
% vectors, one between sms points c to a and one from bounding box point 5
% to 1. We then take the inner product. If it is negative then we do
% comparison (2) otherwise (1).
ca = [sms_x(entry,1)-sms_x(entry+2,1);sms_y(entry,1)-sms_y(entry+2,1)];
normca = norm(ca);
cau = ca/normca;
v51 = [I_beam(i_min).corner(1).trrot(obj_min,1).x2-
I_beam(i_min).corner(5).trrot(obj_min,1).x2;I_beam(i_min).corner(1).trrot(obj
_min,1).y2-I_beam(i_min).corner(5).trrot(obj_min,1).y2];
norm51 = norm(v51);
v51u = v51/norm51;
inner_prod = cau(1)*v51u(1) + cau(2)*v51u(2);
if (inner_prod < 0)
    index(1) = 5;
    index(2) = 6;
    index(3) = 1;
    index(4) = 2;
else
    index(1) = 1;
    index(2) = 2;
    index(3) = 5;
    index(4) = 6;
end
%generate a line of blank cells
fprintf(fid3,' , , , , , , , , , , , , , , , , ,  \n');
% sms point a to scanner point index(1) comparison. First transform to
scanner frame.
sms_pt(1,1) = sms_x(entry,1);
sms_pt(2,1) = sms_y(entry,1);
sms_pt(3,1) = sms_z(entry,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) -
I_beam(i_min).corner(index(1)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) -
I_beam(i_min).corner(index(1)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) -
I_beam(i_min).corner(index(1)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f
, %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min)
.ctr_mass.x,object(obj_min).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'a',scanner_pt(1,1),scanner_pt
(2,1),scanner_pt(3,1),index(1),...
```

```
I_beam(i_min).corner(index(1)).trrot(obj_min,1).x2,I_beam(i_min).corner(index
(1)).trrot(obj_min,1).y2,...

I_beam(i_min).corner(index(1)).trrot(obj_min,1).z2,error_x,error_y,error_z);
% sms point b to scanner point index(2) comparison. First transform to
scanner frame.
sms_pt(1,1) = sms_x(entry+1,1);
sms_pt(2,1) = sms_y(entry+1,1);
sms_pt(3,1) = sms_z(entry+1,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) -
I_beam(i_min).corner(index(2)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) -
I_beam(i_min).corner(index(2)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) -
I_beam(i_min).corner(index(2)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f
, %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min)
.ctr_mass.x,object(obj_min).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'b',scanner_pt(1,1),scanner_pt
(2,1),scanner_pt(3,1),index(2),...

I_beam(i_min).corner(index(2)).trrot(obj_min,1).x2,I_beam(i_min).corner(index
(2)).trrot(obj_min,1).y2,...

I_beam(i_min).corner(index(2)).trrot(obj_min,1).z2,error_x,error_y,error_z);
% sms point c to scanner point index(3) comparison. First transform to
scanner frame.
sms_pt(1,1) = sms_x(entry+2,1);
sms_pt(2,1) = sms_y(entry+2,1);
sms_pt(3,1) = sms_z(entry+2,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) -
I_beam(i_min).corner(index(3)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) -
I_beam(i_min).corner(index(3)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) -
I_beam(i_min).corner(index(3)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f
, %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min)
.ctr_mass.x,object(obj_min).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'c',scanner_pt(1,1),scanner_pt
(2,1),scanner_pt(3,1),index(3),...

I_beam(i_min).corner(index(3)).trrot(obj_min,1).x2,I_beam(i_min).corner(index
(3)).trrot(obj_min,1).y2,...

I_beam(i_min).corner(index(3)).trrot(obj_min,1).z2,error_x,error_y,error_z);
% sms point d to scanner point index(4) comparison. First transform to
scanner frame.
```

```
sms_pt(1,1) = sms_x(entry+3,1);
sms_pt(2,1) = sms_y(entry+3,1);
sms_pt(3,1) = sms_z(entry+3,1);
scanner_pt = Transform*(sms_pt - cg) + cf;
error_x = scanner_pt(1,1) -
I_beam(i_min).corner(index(4)).trrot(obj_min,1).x2;
error_y = scanner_pt(2,1) -
I_beam(i_min).corner(index(4)).trrot(obj_min,1).y2;
error_z = scanner_pt(3,1) -
I_beam(i_min).corner(index(4)).trrot(obj_min,1).z2;
fprintf(fid3,'%s , %s , %f , %f , %f , %f , %f , %s , %f , %f , %f , %d , %f
, %f , %f , %f , %f , %f\n',...

ladar_file,I_beam(i_min).name,comparison_error(i_min,obj_min),object(obj_min)
.ctr_mass.x,object(obj_min).ctr_mass.y,...

object(obj_min).ctr_mass.z,pose_angle(obj_min),'d',scanner_pt(1,1),scanner_pt
(2,1),scanner_pt(3,1),index(4),...

I_beam(i_min).corner(index(4)).trrot(obj_min,1).x2,I_beam(i_min).corner(index
(4)).trrot(obj_min,1).y2,...

I_beam(i_min).corner(index(4)).trrot(obj_min,1).z2,error_x,error_y,error_z);
end %while feof(fid1)==0
fclose(fid1);
fclose(fid3);
```

A.2 POSE TABLES

A.2.1 Pose Tables for Binning Segmentation

Table A.1 gives the computed center-of-data-mass and the estimated orientation relative to the scanner frame as computed by the binning algorithm for each of the scan files. Column 1 of the table gives the name of the scan file of (x, y, z) point cloud points. Note that the first letter of the file name gives the I-beam identifier of the beam scanned. The second column gives the I-beam selected by the binning identification algorithm. If the I-beam letter selected is the same as the leading character of the scan file name then a correct identification is indicated. The next three columns give the estimated center-of-data-mass for the identified I-beam. The units are meters from the scanner coordinate center. We say the center-of-data-mass since this estimate is based on the scan file data and does not necessarily represent the physical center-of-mass of the I-beam. The final column gives the orientation of the I-beam in degrees relative to the negative *X*-axis of the scanner coordinates, which lies in the scan direction.

Table A.2 gives the location of four corner points measured by the SMS system on the top flange of the I-beams as well as the estimates of these points using the binning algorithm. The units are in meters in the scanner coordinate frame. The points are labeled as *a*, *b*, *c*, and *d* and are shown schematically in Figure 4.1. The first two columns are similar to those of Table A. 1. The third column identifies the corner points. The next three columns give the SMS measured values in the scanner coordinate frame. The next three columns give the estimates of the points using the binning algorithm.

Table A.3 gives the errors, in meters, between the SMS measurements in Table A.2 and the coordinates computed by the binning algorithm.

Tables A.4, A.5, and A.6 are formatted the same as in Tables A.1, A.2, and A.3, respectively, but the values in these tables were obtained using the TIN segmentation algorithm.

Table A.1. Binning Method: I-Beam ID, Center of Mass, I-beam Orientation in Scanner Coordinate Frame.

File Name	I-Beam Selected	Location of I-Beam Data Center (m)			Orientation from neg X-axis (º)
		x	y	z	
A0_12gon_1_xyz.txt	I-Beam-A	-3.571	5.791	-1.406	32.209
A0-12gon-2_xyz.txt	I-Beam-A	-3.645	5.752	-1.405	31.744
A0-12gon-3_xyz.txt	I-Beam-A	-3.570	5.800	-1.405	32.355
A0-20gon-1_xyz.txt	I-Beam-A	-3.498	5.836	-1.408	32.223
A0-20gon-2_xyz.txt	I-Beam-A	-3.505	5.837	-1.408	31.893
A0-20gon-3_xyz.txt	I-Beam-A	-3.624	5.762	-1.410	32.258
A0-8gon-1_xyz.txt	I-Beam-A	-3.637	5.751	-1.404	31.986
A0-8gon-2_xyz.txt	I-Beam-A	-3.611	5.767	-1.401	32.159
A0-8gon-3_xyz.txt	I-Beam-A	-3.598	5.779	-1.403	32.219
A30-12gon-1_xyz.txt	I-Beam-A	-3.662	5.723	-1.407	62.420
A30-12gon-2_xyz.txt	I-Beam-A	-3.628	5.751	-1.410	62.585
A30-12gon-3_xyz.txt	I-Beam-A	-3.636	5.753	-1.411	62.680
A30-20gon-1_xyz.txt	I-Beam-A	-3.700	5.618	-1.412	62.143
A30-20gon-2_xyz.txt	I-Beam-A	-3.688	5.641	-1.416	61.660
A30-20gon-3_xyz.txt	I-Beam-B	-3.540	5.912	-1.411	64.015
A30-8gon-1_xyz.txt	I-Beam-A	-3.631	5.748	-1.406	62.550
A30-8gon-2_xyz.txt	I-Beam-A	-3.662	5.700	-1.406	62.511
A30-8gon-3_xyz.txt	I-Beam-A	-3.641	5.735	-1.402	62.312
A45-12gon-1_xyz.txt	I-Beam-A	-3.642	5.700	-1.412	77.288
A45-12gon-2_xyz.txt	I-Beam-A	-3.648	5.668	-1.408	77.598
A45-12gon-3_xyz.txt	I-Beam-A	-3.645	5.682	-1.411	77.788
A45-20gon-1_xyz.txt	I-Beam-B	-3.825	4.900	-1.408	77.487
A45-20gon-2_xyz.txt	I-Beam-B	-3.622	5.772	-1.416	79.507
A45-20gon-3_xyz.txt	I-Beam-B	-3.794	4.975	-1.410	78.280
A45-8gon-1_xyz.txt	I-Beam-A	-3.638	5.718	-1.405	77.781
A45-8gon-2_xyz.txt	I-Beam-A	-3.640	5.698	-1.407	77.733
A45-8gon-3_xyz.txt	I-Beam-A	-3.633	5.739	-1.405	78.038
A60-12gon-1_xyz.txt	I-Beam-A	-3.640	5.658	-1.409	91.967
A60-12gon-2_xyz.txt	I-Beam-B	-1.214	5.807	-1.034	141.523
A60-12gon-3_xyz.txt	I-Beam-A	-3.637	5.624	-1.413	90.990
A60-20gon-1_xyz.txt	I-Beam-B	-3.655	6.066	-1.429	90.000
A60-20gon-2_xyz.txt	I-Beam-B	-3.636	4.993	-1.412	90.000
A60-20gon-3_xyz.txt	I-Beam-B	-3.633	4.920	-1.416	90.000
A60-8gon-1_xyz.txt	I-Beam-A	-3.648	5.880	-1.405	91.518
A60-8gon-2_xyz.txt	I-Beam-A	-3.645	5.781	-1.405	92.171
A60-8gon-3_xyz.txt	I-Beam-A	-3.645	5.772	-1.406	91.415
A90-12gon-1_xyz.txt	I-Beam-B	-4.650	3.224	-1.283	159.762
A90-12gon-2_xyz.txt	I-Beam-B	-1.165	5.794	-0.995	148.996
A90-12gon-3_xyz.txt	I-Beam-B	-4.696	3.254	-1.273	112.711

Table A.1. Continued.

File Name	I-Beam Selected	Location of I-Beam Data Center (m)			Orientation from neg X-axis (º)
		x	y	z	
A90-20gon-1_xyz.txt	I-Beam-B	2.032	-4.747	3.257	NaN
A90-20gon-2_xyz.txt	I-Beam-B	2.451	-4.739	3.254	45.000
A90-20gon-3_xyz.txt	I-Beam-B	-1.163	6.718	-1.252	NaN
A90-8gon-1_xyz.txt	I-Beam-B	-4.710	3.261	-1.326	141.863
A90-8gon-2_xyz.txt	I-Beam-B	-4.684	3.229	-1.295	3.911
A90-8gon-3_xyz.txt	I-Beam-B	-1.153	5.816	-0.975	159.973
B0-12gon-1_xyz.txt	I-Beam-B	-3.615	5.829	-1.440	32.744
B0-12gon-2_xyz.txt	I-Beam-B	-3.624	5.814	-1.438	31.696
B0-12gon-3_xyz.txt	I-Beam-B	-3.613	5.822	-1.440	32.820
B0-20gon-1_xyz.txt	I-Beam-B	-3.651	5.800	-1.448	31.825
B0-20gon-2_xyz.txt	I-Beam-B	-3.636	5.819	-1.446	31.699
B0-20gon-3_xyz.txt	I-Beam-B	-3.671	5.801	-1.441	32.800
B0-8gon-1_xyz.txt	I-Beam-B	-3.629	5.817	-1.430	32.708
B0-8gon-2_xyz.txt	I-Beam-B	-3.622	5.819	-1.429	32.244
B0-8gon-3_xyz.txt	I-Beam-B	-3.614	5.821	-1.429	32.033
B30-12gon-1_xyz.txt	I-Beam-B	-3.669	5.776	-1.437	62.782
B30-12gon-2_xyz.txt	I-Beam-B	-3.689	5.763	-1.436	63.327
B30-12gon-3_xyz.txt	I-Beam-B	-3.703	5.730	-1.434	63.006
B30-20gon-1_xyz.txt	I-Beam-B	-3.685	5.752	-1.445	62.654
B30-20gon-2_xyz.txt	I-Beam-B	-3.666	5.753	-1.445	62.809
B30-20gon-3_xyz.txt	I-Beam-B	-3.308	6.531	-1.479	66.666
B30-8gon-1_xyz.txt	I-Beam-B	-3.684	5.772	-1.433	63.231
B30-8gon-2_xyz.txt	I-Beam-B	-3.674	5.793	-1.429	63.075
B30-8gon-3_xyz.txt	I-Beam-B	-3.690	5.763	-1.433	63.304
B45-12gon-1_xyz.txt	I-Beam-B	-3.696	5.667	-1.442	78.667
B45-12gon-2_xyz.txt	I-Beam-B	-3.704	5.659	-1.441	78.414
B45-12gon-3_xyz.txt	I-Beam-B	-3.684	5.726	-1.440	78.423
B45-20gon-1_xyz.txt	I-Beam-B	-3.737	5.469	-1.453	78.579
B45-20gon-2_xyz.txt	I-Beam-B	-3.822	5.097	-1.442	78.753
B45-20gon-3_xyz.txt	I-Beam-B	-3.813	5.153	-1.444	80.171
B45-8gon-1_xyz.txt	I-Beam-B	-3.695	5.684	-1.437	78.156
B45-8gon-2_xyz.txt	I-Beam-B	-3.695	5.682	-1.434	78.131
B45-8gon-3_xyz.txt	I-Beam-B	-3.690	5.717	-1.435	78.473
B60-12gon-1_xyz.txt	I-Beam-B	-3.675	5.679	-1.446	94.947
B60-12gon-2_xyz.txt	I-Beam-B	-4.666	3.232	-1.245	83.806
B60-12gon-3_xyz.txt	I-Beam-B	-3.639	5.402	-1.453	91.515
B60-20gon-1_xyz.txt	I-Beam-B	-3.656	5.387	-1.452	90.000
B60-20gon-2_xyz.txt	I-Beam-B	-3.632	5.229	-1.466	90.000
B60-20gon-3_xyz.txt	I-Beam-B	-3.625	5.207	-1.462	90.000
B60-8gon-1_xyz.txt	I-Beam-B	-3.670	5.755	-1.445	93.867
B60-8gon-2_xyz.txt	I-Beam-B	-3.677	5.731	-1.442	94.583
B60-8gon-3_xyz.txt	I-Beam-B	-3.676	5.674	-1.442	94.665

Table A.1. Continued.

File Name	I-Beam Selected	Location of I-Beam Data Center (m)			Orientation from neg X- axis (º)
		x	y	z	
B90-12gon-1_xyz.txt	I-Beam-B	-4.666	3.258	-1.260	174.570
B90-12gon-2_xyz.txt	I-Beam-B	-1.224	5.815	-1.012	139.937
B90-12gon-3_xyz.txt	I-Beam-B	-1.383	5.973	-1.196	150.085
B90-20gon-1_xyz.txt	I-Beam-B	-4.650	3.163	-1.108	0.000
B90-20gon-2_xyz.txt	I-Beam-B	-4.655	3.273	-1.385	NaN
B90-20gon-3_xyz.txt	I-Beam-B	-4.655	3.166	-1.094	0.000
B90-8gon-1_xyz.txt	I-Beam-B	-4.664	3.237	-1.270	28.167
B90-8gon-2_xyz.txt	I-Beam-B	-4.605	7.720	-0.839	170.683
B90-8gon-3_xyz.txt	I-Beam-B	-4.687	7.649	-1.059	127.764

Table A.2. SMS measurements of the four flange points and computed coordinates using the binning process.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A0_12gon_1_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.814	4.903	-1.195
A0_12gon_1_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.910	5.055	-1.196
A0_12gon_1_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.234	6.529	-1.197
A0_12gon_1_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.330	6.681	-1.197
A0-12gon-2_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.894	4.875	-1.185
A0-12gon-2_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.988	5.028	-1.185
A0-12gon-2_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.301	6.480	-1.205
A0-12gon-2_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.395	6.632	-1.205
A0-12gon-3_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.811	4.911	-1.190
A0-12gon-3_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.907	5.062	-1.192
A0-12gon-3_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.235	6.543	-1.200
A0-12gon-3_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.331	6.694	-1.202
A0-20gon-1_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.738	4.947	-1.189
A0-20gon-1_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.834	5.099	-1.188
A0-20gon-1_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.159	6.573	-1.208
A0-20gon-1_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.254	6.725	-1.207
A0-20gon-2_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.750	4.954	-1.195
A0-20gon-2_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.845	5.107	-1.193
A0-20gon-2_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.161	6.566	-1.204
A0-20gon-2_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.256	6.718	-1.203
A0-20gon-3_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.864	4.873	-1.190
A0-20gon-3_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.960	5.024	-1.189
A0-20gon-3_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.285	6.500	-1.212
A0-20gon-3_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.381	6.652	-1.211
A0-8gon-1_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.882	4.868	-1.188
A0-8gon-1_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.977	5.020	-1.188
A0-8gon-1_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.295	6.483	-1.200
A0-8gon-1_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.390	6.635	-1.200
A0-8gon-2_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.854	4.880	-1.183
A0-8gon-2_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.949	5.032	-1.184
A0-8gon-2_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.272	6.504	-1.199
A0-8gon-2_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.368	6.656	-1.200

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A0-8gon-3_xyz.txt	I-Beam-A	a	-4.872	4.931	-1.200	-4.841	4.892	-1.187
A0-8gon-3_xyz.txt	I-Beam-A	b	-4.968	5.079	-1.202	-4.936	5.044	-1.189
A0-8gon-3_xyz.txt	I-Beam-A	c	-2.290	6.554	-1.215	-2.261	6.518	-1.198
A0-8gon-3_xyz.txt	I-Beam-A	d	-2.385	6.704	-1.216	-2.356	6.670	-1.200
A30-12gon-1_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.290	4.332	-1.188
A30-12gon-1_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.449	4.415	-1.190
A30-12gon-1_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.878	7.035	-1.205
A30-12gon-1_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.037	7.118	-1.207
A30-12gon-2_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.251	4.358	-1.189
A30-12gon-2_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.410	4.441	-1.190
A30-12gon-2_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.847	7.065	-1.212
A30-12gon-2_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.006	7.148	-1.213
A30-12gon-3_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.257	4.360	-1.189
A30-12gon-3_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.417	4.442	-1.191
A30-12gon-3_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.858	7.069	-1.212
A30-12gon-3_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.017	7.151	-1.213
A30-20gon-1_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.333	4.231	-1.185
A30-20gon-1_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.491	4.315	-1.186
A30-20gon-1_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.908	6.927	-1.219
A30-20gon-1_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.066	7.011	-1.220
A30-20gon-2_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.333	4.258	-1.199
A30-20gon-2_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.491	4.343	-1.200
A30-20gon-2_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.885	6.942	-1.213
A30-20gon-2_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.043	7.027	-1.213
A30-20gon-3_xyz.txt	I-Beam-B	a	-4.292	4.390	-1.203	-3.890	4.898	-1.602
A30-20gon-3_xyz.txt	I-Beam-B	b	-4.448	4.472	-1.203	-4.118	5.010	-1.605
A30-20gon-3_xyz.txt	I-Beam-B	c	-2.869	7.088	-1.217	-2.955	6.816	-1.579
A30-20gon-3_xyz.txt	I-Beam-B	d	-3.026	7.169	-1.215	-3.184	6.927	-1.582
A30-8gon-1_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.254	4.355	-1.187
A30-8gon-1_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.413	4.437	-1.188
A30-8gon-1_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.849	7.061	-1.205
A30-8gon-1_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.008	7.144	-1.205

Table A.2. Continued

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A30-8gon-2_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.285	4.307	-1.186
A30-8gon-2_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.445	4.390	-1.186
A30-8gon-2_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.878	7.012	-1.206
A30-8gon-2_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.037	7.095	-1.206
A30-8gon-3_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.268	4.344	-1.182
A30-8gon-3_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.427	4.427	-1.182
A30-8gon-3_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.851	7.044	-1.204
A30-8gon-3_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.010	7.128	-1.203
A30_12gon_1_xyz.txt	I-Beam-A	a	-4.292	4.390	-1.203	-4.290	4.332	-1.188
A30_12gon_1_xyz.txt	I-Beam-A	b	-4.448	4.472	-1.203	-4.449	4.415	-1.190
A30_12gon_1_xyz.txt	I-Beam-A	c	-2.869	7.088	-1.217	-2.878	7.035	-1.205
A30_12gon_1_xyz.txt	I-Beam-A	d	-3.026	7.169	-1.215	-3.037	7.118	-1.207
A45-12gon-1_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.890	4.193	-1.623
A45-12gon-1_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.065	4.233	-1.623
A45-12gon-1_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.219	7.168	-1.620
A45-12gon-1_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.394	7.208	-1.620
A45-12gon-2_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.889	4.160	-1.196
A45-12gon-2_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.065	4.199	-1.197
A45-12gon-2_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.234	7.139	-1.200
A45-12gon-2_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.410	7.177	-1.201
A45-12gon-3_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.882	4.173	-1.199
A45-12gon-3_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.057	4.211	-1.201
A45-12gon-3_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.237	7.154	-1.202
A45-12gon-3_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.412	7.192	-1.203
A45-20gon-1_xyz.txt	I-Beam-B	a	-3.895	4.250	-1.206	-3.932	3.831	-1.586
A45-20gon-1_xyz.txt	I-Beam-B	b	-4.070	4.286	-1.205	-4.180	3.886	-1.586
A45-20gon-1_xyz.txt	I-Beam-B	c	-3.248	7.229	-1.216	-3.470	5.914	-1.593
A45-20gon-1_xyz.txt	I-Beam-B	d	-3.421	7.266	-1.214	-3.718	5.969	-1.593
A45-20gon-2_xyz.txt	I-Beam-B	a	-3.895	4.250	-1.206	-3.691	4.698	-1.588
A45-20gon-2_xyz.txt	I-Beam-B	b	-4.070	4.286	-1.205	-3.941	4.745	-1.588
A45-20gon-2_xyz.txt	I-Beam-B	c	-3.248	7.229	-1.216	-3.303	6.796	-1.608
A45-20gon-2_xyz.txt	I-Beam-B	d	-3.421	7.266	-1.214	-3.552	6.842	-1.608

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A45-20gon-3_xyz.txt	I-Beam-B	a	-3.895	4.250	-1.206	-3.886	3.905	-1.594
A45-20gon-3_xyz.txt	I-Beam-B	b	-4.070	4.286	-1.205	-4.135	3.957	-1.594
A45-20gon-3_xyz.txt	I-Beam-B	c	-3.248	7.229	-1.216	-3.452	5.994	-1.588
A45-20gon-3_xyz.txt	I-Beam-B	d	-3.421	7.266	-1.214	-3.701	6.046	-1.588
A45-8gon-1_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.875	4.210	-1.187
A45-8gon-1_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.050	4.248	-1.189
A45-8gon-1_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.230	7.191	-1.201
A45-8gon-1_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.405	7.229	-1.203
A45-8gon-2_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.879	4.190	-1.192
A45-8gon-2_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.055	4.228	-1.195
A45-8gon-2_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.231	7.170	-1.199
A45-8gon-2_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.407	7.208	-1.202
A45-8gon-3_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.862	4.230	-1.189
A45-8gon-3_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.038	4.267	-1.190
A45-8gon-3_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.230	7.213	-1.200
A45-8gon-3_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.406	7.251	-1.202
A45_12gon_1_xyz.txt	I-Beam-A	a	-3.895	4.250	-1.206	-3.890	4.193	-1.623
A45_12gon_1_xyz.txt	I-Beam-A	b	-4.070	4.286	-1.205	-4.065	4.233	-1.623
A45_12gon_1_xyz.txt	I-Beam-A	c	-3.248	7.229	-1.216	-3.219	7.168	-1.620
A45_12gon_1_xyz.txt	I-Beam-A	d	-3.421	7.266	-1.214	-3.394	7.208	-1.620
A60-12gon-1_xyz.txt	I-Beam-A	a	-3.521	4.208	-1.211	-3.499	4.137	-1.199
A60-12gon-1_xyz.txt	I-Beam-A	b	-3.697	4.200	-1.210	-3.678	4.131	-1.199
A60-12gon-1_xyz.txt	I-Beam-A	c	-3.638	7.255	-1.210	-3.604	7.185	-1.199
A60-12gon-1_xyz.txt	I-Beam-A	d	-3.815	7.248	-1.207	-3.783	7.179	-1.200
A60-12gon-2_xyz.txt	I-Beam-B	a	-3.521	4.208	-1.211	-0.543	5.526	-0.223
A60-12gon-2_xyz.txt	I-Beam-B	b	-3.697	4.200	-1.210	-0.747	5.401	-0.137
A60-12gon-2_xyz.txt	I-Beam-B	c	-3.638	7.255	-1.210	-1.749	6.484	-1.699
A60-12gon-2_xyz.txt	I-Beam-B	d	-3.815	7.248	-1.207	-1.953	6.359	-1.614
A60-12gon-3_xyz.txt	I-Beam-A	a	-3.521	4.208	-1.211	-3.522	4.101	-1.198
A60-12gon-3_xyz.txt	I-Beam-A	b	-3.697	4.200	-1.210	-3.702	4.098	-1.199
A60-12gon-3_xyz.txt	I-Beam-A	c	-3.638	7.255	-1.210	-3.575	7.150	-1.208
A60-12gon-3_xyz.txt	I-Beam-A	d	-3.815	7.248	-1.207	-3.755	7.147	-1.209

Table A.2. Continued

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A60-20gon-1_xyz.txt	I-Beam-B	a	-3.521	4.208	-1.211	-3.528	4.999	-1.615
A60-20gon-1_xyz.txt	I-Beam-B	b	-3.697	4.200	-1.210	-3.782	4.999	-1.615
A60-20gon-1_xyz.txt	I-Beam-B	c	-3.638	7.255	-1.210	-3.528	7.133	-1.607
A60-20gon-1_xyz.txt	I-Beam-B	d	-3.815	7.248	-1.207	-3.782	7.133	-1.607
A60-20gon-2_xyz.txt	I-Beam-B	a	-3.521	4.208	-1.211	-3.509	3.925	-1.588
A60-20gon-2_xyz.txt	I-Beam-B	b	-3.697	4.200	-1.210	-3.763	3.925	-1.588
A60-20gon-2_xyz.txt	I-Beam-B	c	-3.638	7.255	-1.210	-3.509	6.058	-1.599
A60-20gon-2_xyz.txt	I-Beam-B	d	-3.815	7.248	-1.207	-3.763	6.058	-1.599
A60-20gon-3_xyz.txt	I-Beam-B	a	-3.521	4.208	-1.211	-3.506	3.854	-1.602
A60-20gon-3_xyz.txt	I-Beam-B	b	-3.697	4.200	-1.210	-3.760	3.854	-1.602
A60-20gon-3_xyz.txt	I-Beam-B	c	-3.638	7.255	-1.210	-3.506	5.988	-1.593
A60-20gon-3_xyz.txt	I-Beam-B	d	-3.815	7.248	-1.207	-3.760	5.988	-1.593
A60-8gon-1_xyz.txt	I-Beam-A	a	-3.521	4.208	-1.211	-3.520	4.359	-1.190
A60-8gon-1_xyz.txt	I-Beam-A	b	-3.697	4.200	-1.210	-3.699	4.354	-1.192
A60-8gon-1_xyz.txt	I-Beam-A	c	-3.638	7.255	-1.210	-3.601	7.408	-1.199
A60-8gon-1_xyz.txt	I-Beam-A	d	-3.815	7.248	-1.207	-3.780	7.403	-1.200
A60-8gon-2_xyz.txt	I-Beam-A	a	-3.521	4.208	-1.211	-3.499	4.261	-1.192
A60-8gon-2_xyz.txt	I-Beam-A	b	-3.697	4.200	-1.210	-3.678	4.254	-1.194
A60-8gon-2_xyz.txt	I-Beam-A	c	-3.638	7.255	-1.210	-3.615	7.308	-1.198
A60-8gon-2_xyz.txt	I-Beam-A	d	-3.815	7.248	-1.207	-3.794	7.302	-1.199
A60-8gon-3_xyz.txt	I-Beam-A	a	-3.521	4.208	-1.211	-3.520	4.250	-1.192
A60-8gon-3_xyz.txt	I-Beam-A	b	-3.697	4.200	-1.210	-3.700	4.245	-1.194
A60-8gon-3_xyz.txt	I-Beam-A	c	-3.638	7.255	-1.210	-3.595	7.299	-1.198
A60-8gon-3_xyz.txt	I-Beam-A	d	-3.815	7.248	-1.207	-3.775	7.294	-1.200
A90-12gon-1_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-4.686	3.378	-2.361
A90-12gon-1_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-4.780	3.142	-2.362
A90-12gon-1_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-4.856	3.440	-0.235
A90-12gon-1_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-4.950	3.204	-0.236
A90-12gon-2_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-0.369	5.268	-0.469
A90-12gon-2_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-0.561	5.176	-0.331
A90-12gon-2_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-1.760	6.104	-1.854
A90-12gon-2_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-1.953	6.012	-1.716

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A90-12gon-3_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-4.639	3.443	-0.201
A90-12gon-3_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-4.873	3.343	-0.201
A90-12gon-3_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-4.662	3.499	-2.334
A90-12gon-3_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-4.896	3.399	-2.334
A90-20gon-1_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-4.620	3.439	0.711
A90-20gon-1_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-4.874	3.439	0.711
A90-20gon-1_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-4.620	3.439	-1.422
A90-20gon-1_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-4.874	3.439	-1.422
A90-20gon-2_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-4.521	3.293	-0.063
A90-20gon-2_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-4.701	3.472	-0.063
A90-20gon-2_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-4.521	3.293	-2.196
A90-20gon-2_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-4.701	3.472	-2.196
A90-20gon-3_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-1.266	6.914	-0.185
A90-20gon-3_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-1.014	6.882	-0.185
A90-20gon-3_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-1.266	6.914	-2.319
A90-20gon-3_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-1.014	6.882	-2.319
A90-8gon-1_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-4.661	3.336	-0.240
A90-8gon-1_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-4.776	3.109	-0.249
A90-8gon-1_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-4.964	3.573	-2.338
A90-8gon-1_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-5.078	3.347	-2.347
A90-8gon-2_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-4.520	3.051	-0.233
A90-8gon-2_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-4.435	3.291	-0.236
A90-8gon-2_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-4.590	3.047	-2.365
A90-8gon-2_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-4.505	3.286	-2.368
A90-8gon-3_xyz.txt	I-Beam-B	a	-2.770	4.445	-1.213	-0.540	5.825	-0.074
A90-8gon-3_xyz.txt	I-Beam-B	b	-2.918	4.350	-1.213	-0.739	5.704	0.027
A90-8gon-3_xyz.txt	I-Beam-B	c	-4.381	7.033	-1.207	-1.681	6.241	-1.828
A90-8gon-3_xyz.txt	I-Beam-B	d	-4.532	6.940	-1.204	-1.880	6.119	-1.727
B0-12gon-1_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.443	5.149	-1.239
B0-12gon-1_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.580	5.363	-1.244
B0-12gon-1_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.648	6.303	-1.272
B0-12gon-1_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.786	6.517	-1.277

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B0-12gon-2_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.465	5.147	-1.246
B0-12gon-2_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.598	5.363	-1.248
B0-12gon-2_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.649	6.268	-1.264
B0-12gon-2_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.783	6.484	-1.266
B0-12gon-3_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.442	5.141	-1.252
B0-12gon-3_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.580	5.354	-1.257
B0-12gon-3_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.649	6.297	-1.260
B0-12gon-3_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.787	6.511	-1.265
B0-20gon-1_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.489	5.134	-1.246
B0-20gon-1_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.623	5.349	-1.250
B0-20gon-1_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.677	6.259	-1.282
B0-20gon-1_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.811	6.474	-1.287
B0-20gon-2_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.477	5.155	-1.248
B0-20gon-2_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.611	5.371	-1.255
B0-20gon-2_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.662	6.276	-1.274
B0-20gon-2_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.796	6.492	-1.280
B0-20gon-3_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.500	5.120	-1.245
B0-20gon-3_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.637	5.334	-1.251
B0-20gon-3_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.706	6.276	-1.268
B0-20gon-3_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.844	6.489	-1.274
B0-8gon-1_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.458	5.136	-1.238
B0-8gon-1_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.596	5.350	-1.241
B0-8gon-1_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.663	6.289	-1.256
B0-8gon-1_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.800	6.503	-1.260
B0-8gon-2_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.458	5.147	-1.237
B0-8gon-2_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.593	5.362	-1.244
B0-8gon-2_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.653	6.285	-1.251
B0-8gon-2_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.789	6.500	-1.258
B0-8gon-3_xyz.txt	I-Beam-B	a	-4.486	5.171	-1.262	-4.453	5.152	-1.237
B0-8gon-3_xyz.txt	I-Beam-B	b	-4.618	5.381	-1.258	-4.587	5.367	-1.244
B0-8gon-3_xyz.txt	I-Beam-B	c	-2.673	6.298	-1.274	-2.644	6.284	-1.251
B0-8gon-3_xyz.txt	I-Beam-B	d	-2.807	6.510	-1.270	-2.779	6.499	-1.258

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B30-12gon-1_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.042	4.772	-1.241
B30-12gon-1_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.268	4.888	-1.241
B30-12gon-1_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.067	6.669	-1.270
B30-12gon-1_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.292	6.785	-1.270
B30-12gon-2_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.056	4.756	-1.242
B30-12gon-2_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.283	4.870	-1.246
B30-12gon-2_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.098	6.662	-1.263
B30-12gon-2_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.325	6.776	-1.267
B30-12gon-3_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.077	4.724	-1.245
B30-12gon-3_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.303	4.839	-1.250
B30-12gon-3_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.109	6.625	-1.256
B30-12gon-3_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.335	6.740	-1.261
B30-20gon-1_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.060	4.748	-1.249
B30-20gon-1_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.286	4.864	-1.248
B30-20gon-1_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.080	6.643	-1.278
B30-20gon-1_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.306	6.759	-1.277
B30-20gon-2_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.039	4.749	-1.244
B30-20gon-2_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.265	4.865	-1.244
B30-20gon-2_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.065	6.647	-1.281
B30-20gon-2_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.291	6.763	-1.281
B30-20gon-3_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-3.610	5.509	-1.708
B30-20gon-3_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-3.843	5.610	-1.710
B30-20gon-3_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-2.765	7.466	-1.611
B30-20gon-3_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-2.999	7.567	-1.612
B30-8gon-1_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.052	4.764	-1.241
B30-8gon-1_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.278	4.879	-1.244
B30-8gon-1_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.091	6.669	-1.260
B30-8gon-1_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.317	6.783	-1.262
B30-8gon-2_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.043	4.787	-1.235
B30-8gon-2_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.270	4.902	-1.236
B30-8gon-2_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.077	6.689	-1.258
B30-8gon-2_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.304	6.804	-1.260

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B30-8gon-3_xyz.txt	I-Beam-B	a	-4.065	4.797	-1.265	-4.056	4.756	-1.237
B30-8gon-3_xyz.txt	I-Beam-B	b	-4.288	4.911	-1.257	-4.283	4.870	-1.239
B30-8gon-3_xyz.txt	I-Beam-B	c	-3.078	6.687	-1.273	-3.098	6.662	-1.262
B30-8gon-3_xyz.txt	I-Beam-B	d	-3.301	6.801	-1.266	-3.324	6.776	-1.265
B45-12gon-1_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.782	4.598	-1.252
B45-12gon-1_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.031	4.648	-1.254
B45-12gon-1_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.363	6.690	-1.266
B45-12gon-1_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.612	6.739	-1.268
B45-12gon-2_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.796	4.590	-1.251
B45-12gon-2_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.045	4.641	-1.255
B45-12gon-2_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.368	6.680	-1.263
B45-12gon-2_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.617	6.731	-1.267
B45-12gon-3_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.777	4.656	-1.255
B45-12gon-3_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.026	4.707	-1.259
B45-12gon-3_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.349	6.746	-1.257
B45-12gon-3_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.598	6.797	-1.261
B45-20gon-1_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.823	4.399	-1.267
B45-20gon-1_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.072	4.449	-1.266
B45-20gon-1_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.400	6.490	-1.276
B45-20gon-1_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.649	6.540	-1.275
B45-20gon-2_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.908	4.031	-1.235
B45-20gon-2_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.157	4.080	-1.241
B45-20gon-2_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.492	6.123	-1.281
B45-20gon-2_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.741	6.172	-1.286
B45-20gon-3_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.867	4.081	-1.631
B45-20gon-3_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.118	4.125	-1.635
B45-20gon-3_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.503	6.184	-1.616
B45-20gon-3_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.753	6.227	-1.620
B45-8gon-1_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.792	4.615	-1.250
B45-8gon-1_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.040	4.667	-1.253
B45-8gon-1_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.354	6.703	-1.257
B45-8gon-1_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.603	6.755	-1.261

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B45-8gon-2_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.795	4.613	-1.248
B45-8gon-2_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.043	4.665	-1.255
B45-8gon-2_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.356	6.701	-1.251
B45-8gon-2_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.604	6.753	-1.258
B45-8gon-3_xyz.txt	I-Beam-B	a	-3.797	4.690	-1.267	-3.782	4.648	-1.247
B45-8gon-3_xyz.txt	I-Beam-B	b	-4.043	4.739	-1.258	-4.031	4.699	-1.251
B45-8gon-3_xyz.txt	I-Beam-B	c	-3.352	6.776	-1.272	-3.355	6.738	-1.256
B45-8gon-3_xyz.txt	I-Beam-B	d	-3.598	6.826	-1.265	-3.604	6.789	-1.260
B60-12gon-1_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.459	4.627	-1.262
B60-12gon-1_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.712	4.605	-1.265
B60-12gon-1_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.643	6.753	-1.263
B60-12gon-1_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.896	6.731	-1.266
B60-12gon-2_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-4.449	3.214	-0.177
B60-12gon-2_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-4.578	2.995	-0.185
B60-12gon-2_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-4.441	3.286	-2.309
B60-12gon-2_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-4.570	3.067	-2.317
B60-12gon-3_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.483	4.340	-1.638
B60-12gon-3_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.737	4.333	-1.640
B60-12gon-3_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.540	6.472	-1.630
B60-12gon-3_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.793	6.466	-1.631
B60-20gon-1_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.529	4.318	-1.621
B60-20gon-1_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.783	4.318	-1.621
B60-20gon-1_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.529	6.452	-1.646
B60-20gon-1_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.783	6.452	-1.646
B60-20gon-2_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.505	4.160	-1.634
B60-20gon-2_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.759	4.160	-1.634
B60-20gon-2_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.505	6.293	-1.660
B60-20gon-2_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.759	6.293	-1.660
B60-20gon-3_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.498	4.136	-1.618
B60-20gon-3_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.752	4.136	-1.618
B60-20gon-3_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.498	6.269	-1.670
B60-20gon-3_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.752	6.269	-1.670

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B60-8gon-1_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.471	4.701	-1.638
B60-8gon-1_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.724	4.684	-1.638
B60-8gon-1_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.615	6.830	-1.616
B60-8gon-1_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.868	6.812	-1.616
B60-8gon-2_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.464	4.680	-1.635
B60-8gon-2_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.717	4.660	-1.636
B60-8gon-2_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.635	6.806	-1.612
B60-8gon-2_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.888	6.786	-1.613
B60-8gon-3_xyz.txt	I-Beam-B	a	-3.525	4.664	-1.271	-3.464	4.623	-1.250
B60-8gon-3_xyz.txt	I-Beam-B	b	-3.778	4.650	-1.263	-3.717	4.602	-1.251
B60-8gon-3_xyz.txt	I-Beam-B	c	-3.607	6.795	-1.270	-3.637	6.749	-1.269
B60-8gon-3_xyz.txt	I-Beam-B	d	-3.858	6.783	-1.260	-3.890	6.729	-1.270
B90-12gon-1_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.684	3.449	-2.333
B90-12gon-1_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.806	3.226	-2.340
B90-12gon-1_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-4.843	3.464	-0.205
B90-12gon-1_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-4.965	3.241	-0.213
B90-12gon-2_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-0.573	5.517	-0.190
B90-12gon-2_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-0.778	5.388	-0.114
B90-12gon-2_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-1.748	6.505	-1.672
B90-12gon-2_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-1.952	6.376	-1.596
B90-12gon-3_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-0.900	5.946	-0.220
B90-12gon-3_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-1.128	5.865	-0.142
B90-12gon-3_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-1.717	6.415	-2.134
B90-12gon-3_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-1.945	6.335	-2.057
B90-20gon-1_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.468	3.036	-0.041
B90-20gon-1_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.468	3.290	-0.041
B90-20gon-1_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-4.468	3.036	-2.174
B90-20gon-1_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-4.468	3.290	-2.174
B90-20gon-2_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.528	3.455	-0.318
B90-20gon-2_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.782	3.455	-0.318
B90-20gon-2_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-4.528	3.455	-2.452
B90-20gon-2_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-4.782	3.455	-2.452

Table A.2. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B90-20gon-3_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.837	3.293	-0.027
B90-20gon-3_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.837	3.039	-0.027
B90-20gon-3_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-4.837	3.293	-2.161
B90-20gon-3_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-4.837	3.039	-2.161
B90-8gon-1_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.420	3.231	-2.332
B90-8gon-1_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.582	3.035	-2.338
B90-8gon-1_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-4.466	3.206	-0.199
B90-8gon-1_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-4.628	3.011	-0.205
B90-8gon-2_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.191	7.741	-1.847
B90-8gon-2_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.189	7.489	-1.819
B90-8gon-2_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-5.329	7.928	-0.052
B90-8gon-2_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-5.327	7.676	-0.025
B90-8gon-3_xyz.txt	I-Beam-B	a	-2.991	4.832	-1.273	-4.348	7.409	-2.066
B90-8gon-3_xyz.txt	I-Beam-B	b	-3.203	4.698	-1.265	-4.414	7.650	-2.113
B90-8gon-3_xyz.txt	I-Beam-B	c	-4.109	6.648	-1.267	-4.612	7.749	0.024
B90-8gon-3_xyz.txt	I-Beam-B	d	-4.322	6.513	-1.258	-4.678	7.990	-0.024

Table A.3. Errors between SMS measured locations of four flange points and the computed values using the binning process.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A0_12gon_1_xyz.txt	I-Beam-A	a	-0.058	0.028	-0.005
A0_12gon_1_xyz.txt	I-Beam-A	b	-0.058	0.024	-0.007
A0_12gon_1_xyz.txt	I-Beam-A	c	-0.056	0.025	-0.019
A0_12gon_1_xyz.txt	I-Beam-A	d	-0.056	0.024	-0.019
A0-12gon-2_xyz.txt	I-Beam-A	a	0.022	0.056	-0.015
A0-12gon-2_xyz.txt	I-Beam-A	b	0.020	0.051	-0.017
A0-12gon-2_xyz.txt	I-Beam-A	c	0.011	0.075	-0.011
A0-12gon-2_xyz.txt	I-Beam-A	d	0.010	0.072	-0.011
A0-12gon-3_xyz.txt	I-Beam-A	a	-0.062	0.021	-0.010
A0-12gon-3_xyz.txt	I-Beam-A	b	-0.061	0.016	-0.011
A0-12gon-3_xyz.txt	I-Beam-A	c	-0.055	0.011	-0.016
A0-12gon-3_xyz.txt	I-Beam-A	d	-0.054	0.010	-0.014
A0-20gon-1_xyz.txt	I-Beam-A	a	-0.134	-0.016	-0.010
A0-20gon-1_xyz.txt	I-Beam-A	b	-0.134	-0.020	-0.014
A0-20gon-1_xyz.txt	I-Beam-A	c	-0.131	-0.019	-0.007
A0-20gon-1_xyz.txt	I-Beam-A	d	-0.131	-0.021	-0.009
A0-20gon-2_xyz.txt	I-Beam-A	a	-0.122	-0.023	-0.004
A0-20gon-2_xyz.txt	I-Beam-A	b	-0.123	-0.028	-0.009
A0-20gon-2_xyz.txt	I-Beam-A	c	-0.129	-0.011	-0.011
A0-20gon-2_xyz.txt	I-Beam-A	d	-0.129	-0.014	-0.014
A0-20gon-3_xyz.txt	I-Beam-A	a	-0.009	0.059	-0.009
A0-20gon-3_xyz.txt	I-Beam-A	b	-0.009	0.054	-0.013
A0-20gon-3_xyz.txt	I-Beam-A	c	-0.005	0.054	-0.004
A0-20gon-3_xyz.txt	I-Beam-A	d	-0.004	0.052	-0.005
A0-8gon-1_xyz.txt	I-Beam-A	a	0.009	0.064	-0.011
A0-8gon-1_xyz.txt	I-Beam-A	b	0.008	0.059	-0.014
A0-8gon-1_xyz.txt	I-Beam-A	c	0.005	0.071	-0.015
A0-8gon-1_xyz.txt	I-Beam-A	d	0.005	0.069	-0.016
A0-8gon-2_xyz.txt	I-Beam-A	a	-0.019	0.051	-0.016
A0-8gon-2_xyz.txt	I-Beam-A	b	-0.019	0.046	-0.018
A0-8gon-2_xyz.txt	I-Beam-A	c	-0.018	0.050	-0.017
A0-8gon-2_xyz.txt	I-Beam-A	d	-0.017	0.049	-0.017

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A0-8gon-3_xyz.txt	I-Beam-A	a	-0.032	0.039	-0.013
A0-8gon-3_xyz.txt	I-Beam-A	b	-0.032	0.035	-0.014
A0-8gon-3_xyz.txt	I-Beam-A	c	-0.029	0.036	-0.018
A0-8gon-3_xyz.txt	I-Beam-A	d	-0.029	0.034	-0.017
A30-12gon-1_xyz.txt	I-Beam-A	a	-0.002	0.058	-0.016
A30-12gon-1_xyz.txt	I-Beam-A	b	0.001	0.057	-0.013
A30-12gon-1_xyz.txt	I-Beam-A	c	0.009	0.053	-0.011
A30-12gon-1_xyz.txt	I-Beam-A	d	0.011	0.051	-0.008
A30-12gon-2_xyz.txt	I-Beam-A	a	-0.041	0.032	-0.015
A30-12gon-2_xyz.txt	I-Beam-A	b	-0.038	0.031	-0.013
A30-12gon-2_xyz.txt	I-Beam-A	c	-0.022	0.023	-0.005
A30-12gon-2_xyz.txt	I-Beam-A	d	-0.020	0.021	-0.002
A30-12gon-3_xyz.txt	I-Beam-A	a	-0.035	0.031	-0.014
A30-12gon-3_xyz.txt	I-Beam-A	b	-0.031	0.030	-0.012
A30-12gon-3_xyz.txt	I-Beam-A	c	-0.011	0.019	-0.005
A30-12gon-3_xyz.txt	I-Beam-A	d	-0.009	0.017	-0.002
A30-20gon-1_xyz.txt	I-Beam-A	a	0.040	0.159	-0.018
A30-20gon-1_xyz.txt	I-Beam-A	b	0.043	0.158	-0.017
A30-20gon-1_xyz.txt	I-Beam-A	c	0.039	0.161	0.002
A30-20gon-1_xyz.txt	I-Beam-A	d	0.041	0.158	0.005
A30-20gon-2_xyz.txt	I-Beam-A	a	0.041	0.132	-0.004
A30-20gon-2_xyz.txt	I-Beam-A	b	0.043	0.129	-0.003
A30-20gon-2_xyz.txt	I-Beam-A	c	0.016	0.146	-0.004
A30-20gon-2_xyz.txt	I-Beam-A	d	0.017	0.142	-0.002
A30-20gon-3_xyz.txt	I-Beam-B	a	-0.402	-0.508	0.399
A30-20gon-3_xyz.txt	I-Beam-B	b	-0.330	-0.537	0.402
A30-20gon-3_xyz.txt	I-Beam-B	c	0.086	0.272	0.363
A30-20gon-3_xyz.txt	I-Beam-B	d	0.158	0.241	0.367
A30-8gon-1_xyz.txt	I-Beam-A	a	-0.038	0.036	-0.016
A30-8gon-1_xyz.txt	I-Beam-A	b	-0.035	0.035	-0.015
A30-8gon-1_xyz.txt	I-Beam-A	c	-0.020	0.027	-0.012
A30-8gon-1_xyz.txt	I-Beam-A	d	-0.018	0.025	-0.010

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A30-8gon-2_xyz.txt	I-Beam-A	a	-0.007	0.083	-0.017
A30-8gon-2_xyz.txt	I-Beam-A	b	-0.004	0.082	-0.017
A30-8gon-2_xyz.txt	I-Beam-A	c	0.009	0.076	-0.011
A30-8gon-2_xyz.txt	I-Beam-A	d	0.011	0.074	-0.009
A30-8gon-3_xyz.txt	I-Beam-A	a	-0.024	0.046	-0.021
A30-8gon-3_xyz.txt	I-Beam-A	b	-0.021	0.045	-0.021
A30-8gon-3_xyz.txt	I-Beam-A	c	-0.017	0.044	-0.013
A30-8gon-3_xyz.txt	I-Beam-A	d	-0.015	0.041	-0.012
A30_12gon_1_xyz.txt	I-Beam-A	a	-0.002	0.058	-0.016
A30_12gon_1_xyz.txt	I-Beam-A	b	0.001	0.057	-0.013
A30_12gon_1_xyz.txt	I-Beam-A	c	0.009	0.053	-0.011
A30_12gon_1_xyz.txt	I-Beam-A	d	0.011	0.051	-0.008
A45-12gon-1_xyz.txt	I-Beam-A	a	-0.005	0.057	0.417
A45-12gon-1_xyz.txt	I-Beam-A	b	-0.005	0.053	0.419
A45-12gon-1_xyz.txt	I-Beam-A	c	-0.029	0.061	0.404
A45-12gon-1_xyz.txt	I-Beam-A	d	-0.027	0.058	0.406
A45-12gon-2_xyz.txt	I-Beam-A	a	-0.006	0.090	-0.010
A45-12gon-2_xyz.txt	I-Beam-A	b	-0.005	0.087	-0.007
A45-12gon-2_xyz.txt	I-Beam-A	c	-0.013	0.090	-0.015
A45-12gon-2_xyz.txt	I-Beam-A	d	-0.011	0.089	-0.012
A45-12gon-3_xyz.txt	I-Beam-A	a	-0.014	0.077	-0.007
A45-12gon-3_xyz.txt	I-Beam-A	b	-0.013	0.075	-0.004
A45-12gon-3_xyz.txt	I-Beam-A	c	-0.011	0.075	-0.014
A45-12gon-3_xyz.txt	I-Beam-A	d	-0.009	0.074	-0.011
A45-20gon-1_xyz.txt	I-Beam-B	a	0.037	0.419	0.380
A45-20gon-1_xyz.txt	I-Beam-B	b	0.110	0.400	0.381
A45-20gon-1_xyz.txt	I-Beam-B	c	0.222	1.315	0.377
A45-20gon-1_xyz.txt	I-Beam-B	d	0.297	1.297	0.379
A45-20gon-2_xyz.txt	I-Beam-B	a	-0.204	-0.448	0.382
A45-20gon-2_xyz.txt	I-Beam-B	b	-0.129	-0.459	0.384
A45-20gon-2_xyz.txt	I-Beam-B	c	0.055	0.433	0.392
A45-20gon-2_xyz.txt	I-Beam-B	d	0.132	0.424	0.394

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A45-20gon-3_xyz.txt	I-Beam-B	a	-0.009	0.345	0.388
A45-20gon-3_xyz.txt	I-Beam-B	b	0.065	0.329	0.389
A45-20gon-3_xyz.txt	I-Beam-B	c	0.205	1.235	0.373
A45-20gon-3_xyz.txt	I-Beam-B	d	0.280	1.220	0.374
A45-8gon-1_xyz.txt	I-Beam-A	a	-0.020	0.040	-0.019
A45-8gon-1_xyz.txt	I-Beam-A	b	-0.019	0.038	-0.016
A45-8gon-1_xyz.txt	I-Beam-A	c	-0.018	0.038	-0.014
A45-8gon-1_xyz.txt	I-Beam-A	d	-0.016	0.037	-0.011
A45-8gon-2_xyz.txt	I-Beam-A	a	-0.016	0.060	-0.014
A45-8gon-2_xyz.txt	I-Beam-A	b	-0.015	0.058	-0.010
A45-8gon-2_xyz.txt	I-Beam-A	c	-0.016	0.059	-0.016
A45-8gon-2_xyz.txt	I-Beam-A	d	-0.014	0.058	-0.012
A45-8gon-3_xyz.txt	I-Beam-A	a	-0.033	0.020	-0.017
A45-8gon-3_xyz.txt	I-Beam-A	b	-0.032	0.019	-0.015
A45-8gon-3_xyz.txt	I-Beam-A	c	-0.017	0.016	-0.015
A45-8gon-3_xyz.txt	I-Beam-A	d	-0.015	0.016	-0.012
A45_12gon_1_xyz.txt	I-Beam-A	a	-0.005	0.057	0.417
A45_12gon_1_xyz.txt	I-Beam-A	b	-0.005	0.053	0.419
A45_12gon_1_xyz.txt	I-Beam-A	c	-0.029	0.061	0.404
A45_12gon_1_xyz.txt	I-Beam-A	d	-0.027	0.058	0.406
A60-12gon-1_xyz.txt	I-Beam-A	a	-0.022	0.071	-0.012
A60-12gon-1_xyz.txt	I-Beam-A	b	-0.019	0.069	-0.011
A60-12gon-1_xyz.txt	I-Beam-A	c	-0.034	0.070	-0.011
A60-12gon-1_xyz.txt	I-Beam-A	d	-0.032	0.069	-0.008
A60-12gon-2_xyz.txt	I-Beam-B	a	-2.978	-1.317	-0.988
A60-12gon-2_xyz.txt	I-Beam-B	b	-2.951	-1.201	-1.073
A60-12gon-2_xyz.txt	I-Beam-B	c	-1.889	0.771	0.489
A60-12gon-2_xyz.txt	I-Beam-B	d	-1.862	0.889	0.406
A60-12gon-3_xyz.txt	I-Beam-A	a	0.002	0.107	-0.013
A60-12gon-3_xyz.txt	I-Beam-A	b	0.004	0.102	-0.011
A60-12gon-3_xyz.txt	I-Beam-A	c	-0.063	0.105	-0.002
A60-12gon-3_xyz.txt	I-Beam-A	d	-0.061	0.101	0.002

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A60-20gon-1_xyz.txt	I-Beam-B	a	0.007	-0.791	0.404
A60-20gon-1_xyz.txt	I-Beam-B	b	0.084	-0.800	0.404
A60-20gon-1_xyz.txt	I-Beam-B	c	-0.111	0.122	0.396
A60-20gon-1_xyz.txt	I-Beam-B	d	-0.034	0.115	0.399
A60-20gon-2_xyz.txt	I-Beam-B	a	-0.011	0.284	0.377
A60-20gon-2_xyz.txt	I-Beam-B	b	0.066	0.275	0.378
A60-20gon-2_xyz.txt	I-Beam-B	c	-0.129	1.197	0.389
A60-20gon-2_xyz.txt	I-Beam-B	d	-0.052	1.190	0.392
A60-20gon-3_xyz.txt	I-Beam-B	a	-0.015	0.354	0.391
A60-20gon-3_xyz.txt	I-Beam-B	b	0.063	0.346	0.391
A60-20gon-3_xyz.txt	I-Beam-B	c	-0.132	1.267	0.383
A60-20gon-3_xyz.txt	I-Beam-B	d	-0.055	1.260	0.386
A60-8gon-1_xyz.txt	I-Beam-A	a	-0.001	-0.151	-0.020
A60-8gon-1_xyz.txt	I-Beam-A	b	0.002	-0.154	-0.019
A60-8gon-1_xyz.txt	I-Beam-A	c	-0.037	-0.152	-0.012
A60-8gon-1_xyz.txt	I-Beam-A	d	-0.035	-0.154	-0.007
A60-8gon-2_xyz.txt	I-Beam-A	a	-0.022	-0.053	-0.018
A60-8gon-2_xyz.txt	I-Beam-A	b	-0.019	-0.054	-0.017
A60-8gon-2_xyz.txt	I-Beam-A	c	-0.023	-0.053	-0.013
A60-8gon-2_xyz.txt	I-Beam-A	d	-0.021	-0.053	-0.008
A60-8gon-3_xyz.txt	I-Beam-A	a	-0.001	-0.041	-0.018
A60-8gon-3_xyz.txt	I-Beam-A	b	0.002	-0.046	-0.016
A60-8gon-3_xyz.txt	I-Beam-A	c	-0.043	-0.043	-0.012
A60-8gon-3_xyz.txt	I-Beam-A	d	-0.040	-0.046	-0.007
A90-12gon-1_xyz.txt	I-Beam-B	a	1.916	1.067	1.148
A90-12gon-1_xyz.txt	I-Beam-B	b	1.862	1.209	1.149
A90-12gon-1_xyz.txt	I-Beam-B	c	0.475	3.593	-0.971
A90-12gon-1_xyz.txt	I-Beam-B	d	0.419	3.735	-0.968
A90-12gon-2_xyz.txt	I-Beam-B	a	-2.401	-0.823	-0.744
A90-12gon-2_xyz.txt	I-Beam-B	b	-2.356	-0.826	-0.882
A90-12gon-2_xyz.txt	I-Beam-B	c	-2.621	0.929	0.647
A90-12gon-2_xyz.txt	I-Beam-B	d	-2.579	0.927	0.512

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A90-12gon-3_xyz.txt	I-Beam-B	a	1.869	1.002	-1.012
A90-12gon-3_xyz.txt	I-Beam-B	b	1.955	1.007	-1.012
A90-12gon-3_xyz.txt	I-Beam-B	c	0.281	3.534	1.128
A90-12gon-3_xyz.txt	I-Beam-B	d	0.364	3.541	1.130
A90-20gon-1_xyz.txt	I-Beam-B	a	1.850	1.006	-1.925
A90-20gon-1_xyz.txt	I-Beam-B	b	1.956	0.911	-1.924
A90-20gon-1_xyz.txt	I-Beam-B	c	0.239	3.594	0.216
A90-20gon-1_xyz.txt	I-Beam-B	d	0.342	3.500	0.218
A90-20gon-2_xyz.txt	I-Beam-B	a	1.751	1.152	-1.151
A90-20gon-2_xyz.txt	I-Beam-B	b	1.783	0.878	-1.150
A90-20gon-2_xyz.txt	I-Beam-B	c	0.140	3.740	0.990
A90-20gon-2_xyz.txt	I-Beam-B	d	0.169	3.467	0.992
A90-20gon-3_xyz.txt	I-Beam-B	a	-1.504	-2.469	-1.028
A90-20gon-3_xyz.txt	I-Beam-B	b	-1.904	-2.532	-1.028
A90-20gon-3_xyz.txt	I-Beam-B	c	-3.115	0.119	1.112
A90-20gon-3_xyz.txt	I-Beam-B	d	-3.517	0.057	1.115
A90-8gon-1_xyz.txt	I-Beam-B	a	1.891	1.109	-0.974
A90-8gon-1_xyz.txt	I-Beam-B	b	1.858	1.241	-0.964
A90-8gon-1_xyz.txt	I-Beam-B	c	0.583	3.460	1.132
A90-8gon-1_xyz.txt	I-Beam-B	d	0.547	3.593	1.143
A90-8gon-2_xyz.txt	I-Beam-B	a	1.750	1.394	-0.981
A90-8gon-2_xyz.txt	I-Beam-B	b	1.517	1.060	-0.977
A90-8gon-2_xyz.txt	I-Beam-B	c	0.209	3.987	1.159
A90-8gon-2_xyz.txt	I-Beam-B	d	-0.027	3.654	1.164
A90-8gon-3_xyz.txt	I-Beam-B	a	-2.230	-1.380	-1.139
A90-8gon-3_xyz.txt	I-Beam-B	b	-2.179	-1.353	-1.240
A90-8gon-3_xyz.txt	I-Beam-B	c	-2.700	0.792	0.622
A90-8gon-3_xyz.txt	I-Beam-B	d	-2.652	0.820	0.523
B0-12gon-1_xyz.txt	I-Beam-B	a	-0.043	0.022	-0.023
B0-12gon-1_xyz.txt	I-Beam-B	b	-0.038	0.018	-0.014
B0-12gon-1_xyz.txt	I-Beam-B	c	-0.024	-0.005	-0.002
B0-12gon-1_xyz.txt	I-Beam-B	d	-0.021	-0.007	0.006

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B0-12gon-2_xyz.txt	I-Beam-B	a	-0.021	0.024	-0.016
B0-12gon-2_xyz.txt	I-Beam-B	b	-0.020	0.018	-0.010
B0-12gon-2_xyz.txt	I-Beam-B	c	-0.024	0.030	-0.010
B0-12gon-2_xyz.txt	I-Beam-B	d	-0.024	0.026	-0.005
B0-12gon-3_xyz.txt	I-Beam-B	a	-0.044	0.030	-0.010
B0-12gon-3_xyz.txt	I-Beam-B	b	-0.039	0.027	-0.001
B0-12gon-3_xyz.txt	I-Beam-B	c	-0.024	0.001	-0.014
B0-12gon-3_xyz.txt	I-Beam-B	d	-0.020	-0.001	-0.005
B0-20gon-1_xyz.txt	I-Beam-B	a	0.004	0.037	-0.016
B0-20gon-1_xyz.txt	I-Beam-B	b	0.005	0.032	-0.008
B0-20gon-1_xyz.txt	I-Beam-B	c	0.004	0.040	0.009
B0-20gon-1_xyz.txt	I-Beam-B	d	0.004	0.036	0.016
B0-20gon-2_xyz.txt	I-Beam-B	a	-0.008	0.016	-0.014
B0-20gon-2_xyz.txt	I-Beam-B	b	-0.008	0.010	-0.003
B0-20gon-2_xyz.txt	I-Beam-B	c	-0.011	0.022	0.000
B0-20gon-2_xyz.txt	I-Beam-B	d	-0.011	0.018	0.010
B0-20gon-3_xyz.txt	I-Beam-B	a	0.014	0.051	-0.017
B0-20gon-3_xyz.txt	I-Beam-B	b	0.019	0.048	-0.007
B0-20gon-3_xyz.txt	I-Beam-B	c	0.033	0.022	-0.005
B0-20gon-3_xyz.txt	I-Beam-B	d	0.037	0.020	0.003
B0-8gon-1_xyz.txt	I-Beam-B	a	-0.027	0.035	-0.024
B0-8gon-1_xyz.txt	I-Beam-B	b	-0.023	0.031	-0.017
B0-8gon-1_xyz.txt	I-Beam-B	c	-0.010	0.009	-0.017
B0-8gon-1_xyz.txt	I-Beam-B	d	-0.007	0.007	-0.011
B0-8gon-2_xyz.txt	I-Beam-B	a	-0.028	0.024	-0.025
B0-8gon-2_xyz.txt	I-Beam-B	b	-0.025	0.020	-0.014
B0-8gon-2_xyz.txt	I-Beam-B	c	-0.020	0.013	-0.022
B0-8gon-2_xyz.txt	I-Beam-B	d	-0.018	0.010	-0.012
B0-8gon-3_xyz.txt	I-Beam-B	a	-0.033	0.019	-0.025
B0-8gon-3_xyz.txt	I-Beam-B	b	-0.031	0.014	-0.014
B0-8gon-3_xyz.txt	I-Beam-B	c	-0.029	0.015	-0.023
B0-8gon-3_xyz.txt	I-Beam-B	d	-0.028	0.011	-0.013

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B30-12gon-1_xyz.txt	I-Beam-B	a	-0.023	0.025	-0.024
B30-12gon-1_xyz.txt	I-Beam-B	b	-0.020	0.023	-0.016
B30-12gon-1_xyz.txt	I-Beam-B	c	-0.011	0.018	-0.003
B30-12gon-1_xyz.txt	I-Beam-B	d	-0.008	0.016	0.004
B30-12gon-2_xyz.txt	I-Beam-B	a	-0.009	0.041	-0.023
B30-12gon-2_xyz.txt	I-Beam-B	b	-0.005	0.041	-0.011
B30-12gon-2_xyz.txt	I-Beam-B	c	0.021	0.024	-0.010
B30-12gon-2_xyz.txt	I-Beam-B	d	0.024	0.024	0.001
B30-12gon-3_xyz.txt	I-Beam-B	a	0.012	0.073	-0.021
B30-12gon-3_xyz.txt	I-Beam-B	b	0.015	0.072	-0.008
B30-12gon-3_xyz.txt	I-Beam-B	c	0.031	0.062	-0.018
B30-12gon-3_xyz.txt	I-Beam-B	d	0.034	0.061	-0.005
B30-20gon-1_xyz.txt	I-Beam-B	a	-0.005	0.049	-0.017
B30-20gon-1_xyz.txt	I-Beam-B	b	-0.002	0.046	-0.009
B30-20gon-1_xyz.txt	I-Beam-B	c	0.003	0.044	0.004
B30-20gon-1_xyz.txt	I-Beam-B	d	0.005	0.041	0.011
B30-20gon-2_xyz.txt	I-Beam-B	a	-0.026	0.047	-0.021
B30-20gon-2_xyz.txt	I-Beam-B	b	-0.023	0.045	-0.013
B30-20gon-2_xyz.txt	I-Beam-B	c	-0.013	0.040	0.008
B30-20gon-2_xyz.txt	I-Beam-B	d	-0.010	0.038	0.016
B30-20gon-3_xyz.txt	I-Beam-B	a	-0.456	-0.713	0.443
B30-20gon-3_xyz.txt	I-Beam-B	b	-0.445	-0.699	0.453
B30-20gon-3_xyz.txt	I-Beam-B	c	-0.312	-0.779	0.337
B30-20gon-3_xyz.txt	I-Beam-B	d	-0.302	-0.766	0.347
B30-8gon-1_xyz.txt	I-Beam-B	a	-0.014	0.033	-0.024
B30-8gon-1_xyz.txt	I-Beam-B	b	-0.010	0.032	-0.014
B30-8gon-1_xyz.txt	I-Beam-B	c	0.013	0.018	-0.014
B30-8gon-1_xyz.txt	I-Beam-B	d	0.017	0.017	-0.004
B30-8gon-2_xyz.txt	I-Beam-B	a	-0.022	0.010	-0.031
B30-8gon-2_xyz.txt	I-Beam-B	b	-0.018	0.009	-0.021
B30-8gon-2_xyz.txt	I-Beam-B	c	0.000	-0.002	-0.015
B30-8gon-2_xyz.txt	I-Beam-B	d	0.003	-0.003	-0.006

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m) dx	dy	dz
B30-8gon-3_xyz.txt	I-Beam-B	a	-0.009	0.041	-0.029
B30-8gon-3_xyz.txt	I-Beam-B	b	-0.005	0.041	-0.018
B30-8gon-3_xyz.txt	I-Beam-B	c	0.020	0.025	-0.011
B30-8gon-3_xyz.txt	I-Beam-B	d	0.024	0.025	-0.001
B45-12gon-1_xyz.txt	I-Beam-B	a	-0.015	0.092	-0.014
B45-12gon-1_xyz.txt	I-Beam-B	b	-0.012	0.091	-0.004
B45-12gon-1_xyz.txt	I-Beam-B	c	0.010	0.086	-0.007
B45-12gon-1_xyz.txt	I-Beam-B	d	0.013	0.086	0.003
B45-12gon-2_xyz.txt	I-Beam-B	a	-0.001	0.100	-0.016
B45-12gon-2_xyz.txt	I-Beam-B	b	0.002	0.098	-0.003
B45-12gon-2_xyz.txt	I-Beam-B	c	0.016	0.096	-0.009
B45-12gon-2_xyz.txt	I-Beam-B	d	0.018	0.095	0.003
B45-12gon-3_xyz.txt	I-Beam-B	a	-0.020	0.034	-0.012
B45-12gon-3_xyz.txt	I-Beam-B	b	-0.017	0.032	0.000
B45-12gon-3_xyz.txt	I-Beam-B	c	-0.003	0.029	-0.015
B45-12gon-3_xyz.txt	I-Beam-B	d	-0.001	0.028	-0.004
B45-20gon-1_xyz.txt	I-Beam-B	a	0.026	0.291	0.000
B45-20gon-1_xyz.txt	I-Beam-B	b	0.029	0.290	0.008
B45-20gon-1_xyz.txt	I-Beam-B	c	0.048	0.286	0.003
B45-20gon-1_xyz.txt	I-Beam-B	d	0.051	0.285	0.011
B45-20gon-2_xyz.txt	I-Beam-B	a	0.111	0.660	-0.032
B45-20gon-2_xyz.txt	I-Beam-B	b	0.114	0.659	-0.018
B45-20gon-2_xyz.txt	I-Beam-B	c	0.140	0.653	0.009
B45-20gon-2_xyz.txt	I-Beam-B	d	0.143	0.653	0.022
B45-20gon-3_xyz.txt	I-Beam-B	a	0.070	0.609	0.364
B45-20gon-3_xyz.txt	I-Beam-B	b	0.074	0.614	0.377
B45-20gon-3_xyz.txt	I-Beam-B	c	0.151	0.592	0.344
B45-20gon-3_xyz.txt	I-Beam-B	d	0.155	0.599	0.356
B45-8gon-1_xyz.txt	I-Beam-B	a	-0.005	0.075	-0.017
B45-8gon-1_xyz.txt	I-Beam-B	b	-0.003	0.072	-0.005
B45-8gon-1_xyz.txt	I-Beam-B	c	0.002	0.073	-0.015
B45-8gon-1_xyz.txt	I-Beam-B	d	0.004	0.071	-0.003

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B45-8gon-2_xyz.txt	I-Beam-B	a	-0.002	0.077	-0.019
B45-8gon-2_xyz.txt	I-Beam-B	b	0.000	0.073	-0.004
B45-8gon-2_xyz.txt	I-Beam-B	c	0.004	0.074	-0.022
B45-8gon-2_xyz.txt	I-Beam-B	d	0.006	0.072	-0.007
B45-8gon-3_xyz.txt	I-Beam-B	a	-0.015	0.042	-0.020
B45-8gon-3_xyz.txt	I-Beam-B	b	-0.012	0.040	-0.007
B45-8gon-3_xyz.txt	I-Beam-B	c	0.003	0.037	-0.017
B45-8gon-3_xyz.txt	I-Beam-B	d	0.006	0.036	-0.004
B60-12gon-1_xyz.txt	I-Beam-B	a	-0.066	0.037	-0.010
B60-12gon-1_xyz.txt	I-Beam-B	b	-0.066	0.045	0.002
B60-12gon-1_xyz.txt	I-Beam-B	c	0.036	0.042	-0.007
B60-12gon-1_xyz.txt	I-Beam-B	d	0.038	0.053	0.006
B60-12gon-2_xyz.txt	I-Beam-B	a	0.924	1.450	-1.094
B60-12gon-2_xyz.txt	I-Beam-B	b	0.800	1.656	-1.078
B60-12gon-2_xyz.txt	I-Beam-B	c	0.834	3.509	1.039
B60-12gon-2_xyz.txt	I-Beam-B	d	0.712	3.716	1.057
B60-12gon-3_xyz.txt	I-Beam-B	a	-0.042	0.324	0.367
B60-12gon-3_xyz.txt	I-Beam-B	b	-0.041	0.318	0.377
B60-12gon-3_xyz.txt	I-Beam-B	c	-0.068	0.323	0.360
B60-12gon-3_xyz.txt	I-Beam-B	d	-0.065	0.318	0.371
B60-20gon-1_xyz.txt	I-Beam-B	a	0.004	0.346	0.350
B60-20gon-1_xyz.txt	I-Beam-B	b	0.005	0.332	0.359
B60-20gon-1_xyz.txt	I-Beam-B	c	-0.078	0.343	0.376
B60-20gon-1_xyz.txt	I-Beam-B	d	-0.075	0.332	0.386
B60-20gon-2_xyz.txt	I-Beam-B	a	-0.020	0.504	0.363
B60-20gon-2_xyz.txt	I-Beam-B	b	-0.019	0.491	0.372
B60-20gon-2_xyz.txt	I-Beam-B	c	-0.102	0.502	0.391
B60-20gon-2_xyz.txt	I-Beam-B	d	-0.099	0.490	0.400
B60-20gon-3_xyz.txt	I-Beam-B	a	-0.027	0.528	0.346
B60-20gon-3_xyz.txt	I-Beam-B	b	-0.026	0.515	0.355
B60-20gon-3_xyz.txt	I-Beam-B	c	-0.109	0.526	0.400
B60-20gon-3_xyz.txt	I-Beam-B	d	-0.107	0.515	0.410

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B60-8gon-1_xyz.txt	I-Beam-B	a	-0.054	-0.037	0.367
B60-8gon-1_xyz.txt	I-Beam-B	b	-0.053	-0.033	0.376
B60-8gon-1_xyz.txt	I-Beam-B	c	0.008	-0.034	0.346
B60-8gon-1_xyz.txt	I-Beam-B	d	0.010	-0.029	0.356
B60-8gon-2_xyz.txt	I-Beam-B	a	-0.061	-0.016	0.364
B60-8gon-2_xyz.txt	I-Beam-B	b	-0.060	-0.009	0.373
B60-8gon-2_xyz.txt	I-Beam-B	c	0.028	-0.011	0.342
B60-8gon-2_xyz.txt	I-Beam-B	d	0.030	-0.003	0.353
B60-8gon-3_xyz.txt	I-Beam-B	a	-0.061	0.041	-0.022
B60-8gon-3_xyz.txt	I-Beam-B	b	-0.061	0.048	-0.012
B60-8gon-3_xyz.txt	I-Beam-B	c	0.030	0.046	-0.001
B60-8gon-3_xyz.txt	I-Beam-B	d	0.032	0.055	0.010
B90-12gon-1_xyz.txt	I-Beam-B	a	1.693	1.384	1.060
B90-12gon-1_xyz.txt	I-Beam-B	b	1.603	1.472	1.076
B90-12gon-1_xyz.txt	I-Beam-B	c	0.734	3.184	-1.062
B90-12gon-1_xyz.txt	I-Beam-B	d	0.643	3.272	-1.046
B90-12gon-2_xyz.txt	I-Beam-B	a	-2.418	-0.685	-1.083
B90-12gon-2_xyz.txt	I-Beam-B	b	-2.425	-0.690	-1.150
B90-12gon-2_xyz.txt	I-Beam-B	c	-2.361	0.143	0.405
B90-12gon-2_xyz.txt	I-Beam-B	d	-2.370	0.138	0.338
B90-12gon-3_xyz.txt	I-Beam-B	a	-2.091	-1.113	-1.053
B90-12gon-3_xyz.txt	I-Beam-B	b	-2.075	-1.167	-1.122
B90-12gon-3_xyz.txt	I-Beam-B	c	-2.392	0.233	0.867
B90-12gon-3_xyz.txt	I-Beam-B	d	-2.377	0.178	0.798
B90-20gon-1_xyz.txt	I-Beam-B	a	1.477	1.796	-1.232
B90-20gon-1_xyz.txt	I-Beam-B	b	1.265	1.408	-1.224
B90-20gon-1_xyz.txt	I-Beam-B	c	0.359	3.612	0.907
B90-20gon-1_xyz.txt	I-Beam-B	d	0.146	3.223	0.916
B90-20gon-2_xyz.txt	I-Beam-B	a	1.537	1.377	-0.955
B90-20gon-2_xyz.txt	I-Beam-B	b	1.579	1.243	-0.947
B90-20gon-2_xyz.txt	I-Beam-B	c	0.420	3.193	1.185
B90-20gon-2_xyz.txt	I-Beam-B	d	0.460	3.059	1.193

Table A.3. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B90-20gon-3_xyz.txt	I-Beam-B	a	1.845	1.539	-1.246
B90-20gon-3_xyz.txt	I-Beam-B	b	1.633	1.659	-1.238
B90-20gon-3_xyz.txt	I-Beam-B	c	0.728	3.355	0.894
B90-20gon-3_xyz.txt	I-Beam-B	d	0.514	3.475	0.902
B90-8gon-1_xyz.txt	I-Beam-B	a	1.429	1.601	1.059
B90-8gon-1_xyz.txt	I-Beam-B	b	1.379	1.663	1.073
B90-8gon-1_xyz.txt	I-Beam-B	c	0.357	3.442	-1.068
B90-8gon-1_xyz.txt	I-Beam-B	d	0.306	3.503	-1.054
B90-8gon-2_xyz.txt	I-Beam-B	a	1.200	-2.909	0.574
B90-8gon-2_xyz.txt	I-Beam-B	b	0.986	-2.791	0.555
B90-8gon-2_xyz.txt	I-Beam-B	c	1.220	-1.280	-1.215
B90-8gon-2_xyz.txt	I-Beam-B	d	1.005	-1.162	-1.234
B90-8gon-3_xyz.txt	I-Beam-B	a	1.357	-2.577	0.793
B90-8gon-3_xyz.txt	I-Beam-B	b	1.211	-2.952	0.849
B90-8gon-3_xyz.txt	I-Beam-B	c	0.503	-1.101	-1.291
B90-8gon-3_xyz.txt	I-Beam-B	d	0.356	-1.476	-1.235

A.2.1 Pose Tables for TIN Segmentation

Table A.4. I-beam segmented using TIN procedure, its center-of-scan-data-mass, and the final orientation relative to the scanner.

File Name	I-Beam Selected	Location of Data Center (m)			Orientation from neg. X-axis (°)
		x	y	z	
A0-12gon-1_xyz.seg	I-Beam-A	-3.590	5.772	-1.418	32.239
A0-12gon-2_xyz.seg	I-Beam-A	-3.586	5.775	-1.418	32.280
A0-12gon-3_xyz.seg	I-Beam-A	-3.594	5.770	-1.418	32.266
A0-20gon-1_xyz.seg	I-Beam-A	-3.604	5.765	-1.420	32.096
A0-20gon-2_xyz.seg	I-Beam-A	-3.591	5.775	-1.421	32.189
A0-20gon-3_xyz.seg	I-Beam-A	-3.603	5.766	-1.421	32.024
A0-8gon-1_xyz.seg	I-Beam-A	-3.594	5.764	-1.418	32.241
A0-8gon-2_xyz.seg	I-Beam-A	-3.594	5.767	-1.418	32.291
A0-8gon-3_xyz.seg	I-Beam-A	-3.597	5.769	-1.417	32.279
A30-12gon-1_xyz.seg	I-Beam-A	-3.706	5.592	-1.421	62.233
A30-12gon-2_xyz.seg	I-Beam-A	-3.709	5.589	-1.420	62.235
A30-12gon-3_xyz.seg	I-Beam-A	-3.706	5.594	-1.422	62.194
A30-20gon-1_xyz.seg	I-Beam-A	-3.706	5.599	-1.419	62.225
A30-20gon-2_xyz.seg	I-Beam-A	-3.713	5.579	-1.425	62.341
A30-20gon-3_xyz.seg	I-Beam-A	-3.708	5.590	-1.425	62.299
A30-8gon-1_xyz.seg	I-Beam-A	-3.710	5.584	-1.419	62.206
A30-8gon-2_xyz.seg	I-Beam-A	-3.706	5.590	-1.420	62.206
A30-8gon-3_xyz.seg	I-Beam-A	-3.707	5.593	-1.418	62.169
A45-12gon-1_xyz.seg	I-Beam-A	-3.680	5.473	-1.423	77.583
A45-12gon-2_xyz.seg	I-Beam-A	-3.681	5.468	-1.423	77.679
A45-12gon-3_xyz.seg	I-Beam-A	-3.681	5.476	-1.422	77.610
A45-20gon-1_xyz.seg	I-Beam-A	-3.678	5.491	-1.423	77.517
A45-20gon-2_xyz.seg	I-Beam-A	-3.682	5.466	-1.426	77.750
A45-20gon-3_xyz.seg	I-Beam-A	-3.680	5.466	-1.428	77.520
A45-8gon-1_xyz.seg	I-Beam-A	-3.679	5.474	-1.421	77.563
A45-8gon-2_xyz.seg	I-Beam-A	-3.679	5.472	-1.421	77.570
A45-8gon-3_xyz.seg	I-Beam-A	-3.679	5.476	-1.420	77.538
A60-12gon-1_xyz.seg	I-Beam-A	-3.618	5.382	-1.425	91.906
A60-12gon-2_xyz.seg	I-Beam-A	-3.618	5.376	-1.427	91.869
A60-12gon-3_xyz.seg	I-Beam-A	-3.618	5.382	-1.426	91.938
A60-20gon-1_xyz.seg	I-Beam-A	-3.618	5.361	-1.429	92.061
A60-20gon-2_xyz.seg	I-Beam-A	-3.618	5.367	-1.428	92.052
A60-20gon-3_xyz.seg	I-Beam-A	-3.617	5.360	-1.428	92.054
A60-8gon-1_xyz.seg	I-Beam-A	-3.616	5.364	-1.425	91.985
A60-8gon-2_xyz.seg	I-Beam-A	-3.617	5.368	-1.423	91.997
A60-8gon-3_xyz.seg	I-Beam-A	-3.618	5.361	-1.423	91.986

Table A.4. Continued.

File Name	I-Beam Selected	Location of Data Center (m)			Orientation from neg X-axis (º)
		x	y	z	
A90-12gon-1_xyz.seg	I-Beam-B	-3.080	4.752	-1.470	119.141
A90-12gon-2_xyz.seg	I-Beam-B	-1.135	6.056	-1.179	87.410
A90-12gon-3_xyz.seg	I-Beam-B	-3.096	4.813	-1.456	121.734
A90-20gon-1_xyz.seg	I-Beam-B	-3.138	4.879	-1.453	119.925
A90-20gon-2_xyz.seg	I-Beam-A	-3.195	4.981	-1.446	120.459
A90-20gon-3_xyz.seg	I-Beam-B	-3.156	4.906	-1.436	121.547
A90-8gon-1_xyz.seg	I-Beam-B	-4.693	7.692	-1.085	99.791
A90-8gon-2_xyz.seg	I-Beam-B	-4.696	7.699	-1.116	89.177
A90-8gon-3_xyz.seg	I-Beam-B	-1.136	6.121	-1.186	87.804
B0-12gon-1_xyz.seg	I-Beam-B	-3.613	5.793	-1.453	32.270
B0-12gon-2_xyz.seg	I-Beam-B	-3.620	5.789	-1.454	32.302
B0-12gon-3_xyz.seg	I-Beam-B	-3.614	5.793	-1.454	32.127
B0-20gon-1_xyz.seg	I-Beam-B	-3.608	5.800	-1.456	32.426
B0-20gon-2_xyz.seg	I-Beam-B	-3.618	5.791	-1.459	32.353
B0-20gon-3_xyz.seg	I-Beam-B	-3.607	5.799	-1.457	32.480
B0-8gon-1_xyz.seg	I-Beam-B	-3.610	5.792	-1.453	32.163
B0-8gon-2_xyz.seg	I-Beam-B	-3.612	5.790	-1.452	32.352
B0-8gon-3_xyz.seg	I-Beam-B	-3.612	5.791	-1.452	32.125
B30-12gon-1_xyz.seg	I-Beam-B	-3.689	5.684	-1.455	62.186
B30-12gon-2_xyz.seg	I-Beam-B	-3.691	5.682	-1.454	62.547
B30-12gon-3_xyz.seg	I-Beam-B	-3.690	5.686	-1.452	62.168
B30-20gon-1_xyz.seg	I-Beam-B	-3.688	5.691	-1.456	62.442
B30-20gon-2_xyz.seg	I-Beam-B	-3.693	5.676	-1.457	62.545
B30-20gon-3_xyz.seg	I-Beam-B	-3.689	5.684	-1.458	62.456
B30-8gon-1_xyz.seg	I-Beam-B	-3.687	5.688	-1.451	62.196
B30-8gon-2_xyz.seg	I-Beam-B	-3.691	5.680	-1.452	62.462
B30-8gon-3_xyz.seg	I-Beam-B	-3.686	5.688	-1.454	62.144
B45-12gon-1_xyz.seg	I-Beam-B	-3.688	5.596	-1.456	77.521
B45-12gon-2_xyz.seg	I-Beam-B	-3.689	5.593	-1.456	77.721
B45-12gon-3_xyz.seg	I-Beam-B	-3.690	5.592	-1.457	77.594
B45-20gon-1_xyz.seg	I-Beam-B	-3.691	5.580	-1.463	77.769
B45-20gon-2_xyz.seg	I-Beam-B	-3.694	5.585	-1.456	77.522
B45-20gon-3_xyz.seg	I-Beam-B	-3.694	5.577	-1.462	77.210
B45-8gon-1_xyz.seg	I-Beam-B	-3.686	5.594	-1.456	77.388
B45-8gon-2_xyz.seg	I-Beam-B	-3.687	5.594	-1.454	77.587
B45-8gon-3_xyz.seg	I-Beam-B	-3.688	5.589	-1.454	77.408
B60-12gon-1_xyz.seg	I-Beam-B	-3.645	5.516	-1.460	91.240
B60-12gon-2_xyz.seg	I-Beam-B	-3.643	5.514	-1.461	91.531
B60-12gon-3_xyz.seg	I-Beam-B	-3.645	5.510	-1.461	91.531

Table A.4. Continued.

File Name	I-Beam Selected	Location of Data Center (m)			Orientation from neg X-axis (º)
		x	y	z	
B60-20gon-1_xyz.seg	I-Beam-B	-3.645	5.499	-1.462	91.598
B60-20gon-2_xyz.seg	I-Beam-B	-3.643	5.522	-1.463	91.193
B60-20gon-3_xyz.seg	I-Beam-B	-3.643	5.519	-1.465	91.509
B60-8gon-1_xyz.seg	I-Beam-A	-3.641	5.512	-1.462	91.414
B60-8gon-2_xyz.seg	I-Beam-B	-3.644	5.513	-1.458	91.301
B60-8gon-3_xyz.seg	I-Beam-A	-3.644	5.513	-1.462	91.445
B90-12gon-1_xyz.seg	I-Beam-B	-3.366	5.203	-1.460	120.925
B90-12gon-2_xyz.seg	I-Beam-B	-3.367	5.215	-1.464	120.406
B90-12gon-3_xyz.seg	I-Beam-B	-3.363	5.200	-1.459	120.570
B90-20gon-1_xyz.seg	I-Beam-B	-3.344	5.166	-1.470	121.639
B90-20gon-2_xyz.seg	I-Beam-B	-3.386	5.239	-1.460	121.126
B90-20gon-3_xyz.seg	I-Beam-B	-4.802	3.305	-1.352	146.149
B90-8gon-1_xyz.seg	I-Beam-B	-3.400	5.242	-1.467	122.684
B90-8gon-2_xyz.seg	I-Beam-B	-3.380	5.222	-1.466	120.996
B90-8gon-3_xyz.seg	I-Beam-B	-3.381	5.232	-1.467	120.866

Table A.5. SMS measurements of the four flange points and the computed values using the TIN segmentation process.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A0-12gon-1_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.836	4.891	-1.197
A0-12gon-1_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.932	5.043	-1.206
A0-12gon-1_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.256	6.518	-1.212
A0-12gon-1_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.352	6.670	-1.220
A0-12gon-2_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.832	4.893	-1.200
A0-12gon-2_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.927	5.044	-1.208
A0-12gon-2_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.253	6.521	-1.210
A0-12gon-2_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.349	6.673	-1.218
A0-12gon-3_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.839	4.888	-1.198
A0-12gon-3_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.935	5.039	-1.205
A0-12gon-3_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.261	6.516	-1.212
A0-12gon-3_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.356	6.667	-1.220
A0-20gon-1_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.850	4.884	-1.202
A0-20gon-1_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.945	5.036	-1.207
A0-20gon-1_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.267	6.504	-1.214
A0-20gon-1_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.362	6.656	-1.219
A0-20gon-2_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.837	4.893	-1.204
A0-20gon-2_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.932	5.045	-1.210
A0-20gon-2_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.256	6.518	-1.214
A0-20gon-2_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.351	6.670	-1.220
A0-20gon-3_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.851	4.887	-1.201
A0-20gon-3_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.946	5.039	-1.206
A0-20gon-3_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.265	6.504	-1.217
A0-20gon-3_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.361	6.656	-1.222
A0-8gon-1_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.843	4.886	-1.198
A0-8gon-1_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.938	5.038	-1.209
A0-8gon-1_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.263	6.513	-1.208
A0-8gon-1_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.359	6.665	-1.219

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A0-8gon-2_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.842	4.888	-1.199
A0-8gon-2_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.938	5.040	-1.211
A0-8gon-2_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.264	6.517	-1.207
A0-8gon-2_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.360	6.669	-1.219
A0-8gon-3_xyz.seg	I-Beam-A	a	-4.866	4.924	-1.192	-4.845	4.889	-1.200
A0-8gon-3_xyz.seg	I-Beam-A	b	-4.962	5.071	-1.195	-4.940	5.040	-1.210
A0-8gon-3_xyz.seg	I-Beam-A	c	-2.283	6.545	-1.222	-2.266	6.518	-1.206
A0-8gon-3_xyz.seg	I-Beam-A	d	-2.378	6.695	-1.223	-2.362	6.669	-1.216
A30-12gon-1_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.345	4.206	-1.202
A30-12gon-1_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.503	4.290	-1.210
A30-12gon-1_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.924	6.905	-1.214
A30-12gon-1_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.083	6.988	-1.221
A30-12gon-2_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.347	4.203	-1.202
A30-12gon-2_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.505	4.286	-1.210
A30-12gon-2_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.926	6.901	-1.213
A30-12gon-2_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.085	6.985	-1.220
A30-12gon-3_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.345	4.208	-1.200
A30-12gon-3_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.503	4.292	-1.208
A30-12gon-3_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.922	6.906	-1.217
A30-12gon-3_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.081	6.989	-1.224
A30-20gon-1_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.340	4.211	-1.199
A30-20gon-1_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.499	4.295	-1.203
A30-20gon-1_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.919	6.909	-1.217
A30-20gon-1_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.078	6.993	-1.221
A30-20gon-2_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.349	4.192	-1.206
A30-20gon-2_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.508	4.275	-1.214
A30-20gon-2_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.933	6.893	-1.217
A30-20gon-2_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.092	6.976	-1.225
A30-20gon-3_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.347	4.204	-1.207
A30-20gon-3_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.505	4.287	-1.216
A30-20gon-3_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.929	6.904	-1.215
A30-20gon-3_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.088	6.987	-1.224

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A30-8gon-1_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.351	4.198	-1.202
A30-8gon-1_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.510	4.282	-1.211
A30-8gon-1_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.929	6.896	-1.209
A30-8gon-1_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.088	6.980	-1.218
A30-8gon-2_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.348	4.206	-1.199
A30-8gon-2_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.506	4.289	-1.209
A30-8gon-2_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.926	6.903	-1.212
A30-8gon-2_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.084	6.987	-1.222
A30-8gon-3_xyz.seg	I-Beam-A	a	-4.287	4.383	-1.197	-4.348	4.208	-1.200
A30-8gon-3_xyz.seg	I-Beam-A	b	-4.443	4.465	-1.196	-4.506	4.292	-1.208
A30-8gon-3_xyz.seg	I-Beam-A	c	-2.861	7.079	-1.222	-2.924	6.905	-1.210
A30-8gon-3_xyz.seg	I-Beam-A	d	-3.018	7.160	-1.220	-3.082	6.988	-1.218
A45-12gon-1_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-3.929	3.967	-1.209
A45-12gon-1_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-4.104	4.005	-1.217
A45-12gon-1_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.273	6.945	-1.211
A45-12gon-1_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.448	6.983	-1.218
A45-12gon-2_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-4.102	3.999	-1.219
A45-12gon-2_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-3.927	3.961	-1.212
A45-12gon-2_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.452	6.978	-1.216
A45-12gon-2_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.277	6.940	-1.209
A45-12gon-3_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-3.928	3.969	-1.208
A45-12gon-3_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-4.103	4.008	-1.215
A45-12gon-3_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.273	6.948	-1.209
A45-12gon-3_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.448	6.986	-1.216
A45-20gon-1_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-3.928	3.986	-1.203
A45-20gon-1_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-4.103	4.024	-1.210
A45-20gon-1_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.268	6.963	-1.217
A45-20gon-1_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.443	7.002	-1.224
A45-20gon-2_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-4.102	3.996	-1.222
A45-20gon-2_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-3.927	3.958	-1.214
A45-20gon-2_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.455	6.976	-1.220
A45-20gon-2_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.280	6.938	-1.212

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A45-20gon-3_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-3.936	3.961	-1.209
A45-20gon-3_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-4.111	4.000	-1.221
A45-20gon-3_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.277	6.939	-1.216
A45-20gon-3_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.452	6.977	-1.228
A45-8gon-1_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-3.930	3.968	-1.206
A45-8gon-1_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-4.105	4.007	-1.215
A45-8gon-1_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.273	6.946	-1.208
A45-8gon-1_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.448	6.985	-1.216
A45-8gon-2_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-4.105	4.004	-1.216
A45-8gon-2_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-3.930	3.965	-1.208
A45-8gon-2_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.448	6.982	-1.215
A45-8gon-2_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.273	6.943	-1.207
A45-8gon-3_xyz.seg	I-Beam-A	a	-3.890	4.242	-1.201	-3.930	3.970	-1.205
A45-8gon-3_xyz.seg	I-Beam-A	b	-4.064	4.278	-1.200	-4.105	4.009	-1.213
A45-8gon-3_xyz.seg	I-Beam-A	c	-3.240	7.220	-1.220	-3.272	6.948	-1.208
A45-8gon-3_xyz.seg	I-Beam-A	d	-3.413	7.258	-1.217	-3.447	6.986	-1.217
A60-12gon-1_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.664	3.854	-1.222
A60-12gon-1_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.485	3.860	-1.216
A60-12gon-1_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.765	6.902	-1.216
A60-12gon-1_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.586	6.908	-1.210
A60-12gon-2_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.664	3.849	-1.222
A60-12gon-2_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.485	3.855	-1.217
A60-12gon-2_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.763	6.897	-1.218
A60-12gon-2_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.584	6.903	-1.213
A60-12gon-3_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.664	3.855	-1.223
A60-12gon-3_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.485	3.861	-1.216
A60-12gon-3_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.768	6.903	-1.218
A60-12gon-3_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.588	6.909	-1.211
A60-20gon-1_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.661	3.833	-1.226
A60-20gon-1_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.481	3.839	-1.220
A60-20gon-1_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.770	6.880	-1.219
A60-20gon-1_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.591	6.887	-1.212

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A60-20gon-2_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.658	3.840	-1.225
A60-20gon-2_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.479	3.846	-1.220
A60-20gon-2_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.768	6.887	-1.217
A60-20gon-2_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.588	6.894	-1.213
A60-20gon-3_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.661	3.832	-1.228
A60-20gon-3_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.482	3.838	-1.220
A60-20gon-3_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.770	6.879	-1.217
A60-20gon-3_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.591	6.886	-1.210
A60-8gon-1_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.662	3.836	-1.225
A60-8gon-1_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.483	3.842	-1.217
A60-8gon-1_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.768	6.884	-1.215
A60-8gon-1_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.589	6.890	-1.207
A60-8gon-2_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.661	3.841	-1.222
A60-8gon-2_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.482	3.847	-1.216
A60-8gon-2_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.767	6.888	-1.212
A60-8gon-2_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.588	6.895	-1.206
A60-8gon-3_xyz.seg	I-Beam-A	a	-3.515	4.200	-1.207	-3.662	3.833	-1.223
A60-8gon-3_xyz.seg	I-Beam-A	b	-3.692	4.192	-1.206	-3.482	3.839	-1.217
A60-8gon-3_xyz.seg	I-Beam-A	c	-3.630	7.247	-1.213	-3.767	6.881	-1.211
A60-8gon-3_xyz.seg	I-Beam-A	d	-3.807	7.240	-1.209	-3.588	6.887	-1.205
A90-12gon-1_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-2.468	3.900	-1.763
A90-12gon-1_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-2.689	3.775	-1.754
A90-12gon-1_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-3.501	5.754	-1.547
A90-12gon-1_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-3.723	5.629	-1.538
A90-12gon-2_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-0.904	7.106	-1.359
A90-12gon-2_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-1.139	7.142	-1.271
A90-12gon-2_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-0.996	5.060	-0.761
A90-12gon-2_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-1.232	5.096	-0.673
A90-12gon-3_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-2.432	3.990	-1.718
A90-12gon-3_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-2.648	3.857	-1.724
A90-12gon-3_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-3.551	5.799	-1.549
A90-12gon-3_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-3.767	5.666	-1.555

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
A90-20gon-1_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-2.499	4.028	-1.687
A90-20gon-1_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-2.719	3.902	-1.692
A90-20gon-1_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-3.562	5.875	-1.577
A90-20gon-1_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-3.782	5.748	-1.582
A90-20gon-2_xyz.seg	I-Beam-A	a	-2.765	4.436	-1.214	-2.345	3.725	-1.722
A90-20gon-2_xyz.seg	I-Beam-A	b	-2.912	4.341	-1.213	-2.500	3.634	-1.727
A90-20gon-2_xyz.seg	I-Beam-A	c	-4.373	7.025	-1.206	-3.889	6.351	-1.584
A90-20gon-2_xyz.seg	I-Beam-A	d	-4.524	6.932	-1.202	-4.044	6.260	-1.589
A90-20gon-3_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-2.493	4.083	-1.701
A90-20gon-3_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-2.710	3.951	-1.709
A90-20gon-3_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-3.606	5.895	-1.524
A90-20gon-3_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-3.823	5.763	-1.533
A90-8gon-1_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-4.512	7.689	-2.159
A90-8gon-1_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-4.462	7.443	-2.120
A90-8gon-1_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-4.568	8.013	-0.051
A90-8gon-1_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-4.518	7.767	-0.012
A90-8gon-2_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-4.543	8.003	-0.081
A90-8gon-2_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-4.490	7.757	-0.048
A90-8gon-2_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-4.547	7.715	-2.195
A90-8gon-2_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-4.494	7.469	-2.162
A90-8gon-3_xyz.seg	I-Beam-B	a	-2.765	4.436	-1.214	-1.147	7.210	-1.149
A90-8gon-3_xyz.seg	I-Beam-B	b	-2.912	4.341	-1.213	-0.911	7.186	-1.241
A90-8gon-3_xyz.seg	I-Beam-B	c	-4.373	7.025	-1.206	-1.227	5.107	-0.797
A90-8gon-3_xyz.seg	I-Beam-B	d	-4.524	6.932	-1.202	-0.991	5.083	-0.888
B0-12gon-1_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.595	5.351	-1.284
B0-12gon-1_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.460	5.139	-1.249
B0-12gon-1_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.791	6.491	-1.295
B0-12gon-1_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.656	6.278	-1.261
B0-12gon-2_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.602	5.347	-1.285
B0-12gon-2_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.467	5.134	-1.251
B0-12gon-2_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.798	6.487	-1.298
B0-12gon-2_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.664	6.274	-1.263

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B0-12gon-3_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.599	5.358	-1.288
B0-12gon-3_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.466	5.146	-1.246
B0-12gon-3_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.792	6.492	-1.303
B0-12gon-3_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.659	6.280	-1.261
B0-20gon-1_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.592	5.360	-1.296
B0-20gon-1_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.458	5.149	-1.253
B0-20gon-1_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.791	6.504	-1.300
B0-20gon-1_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.657	6.293	-1.257
B0-20gon-2_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.600	5.348	-1.291
B0-20gon-2_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.465	5.136	-1.256
B0-20gon-2_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.797	6.490	-1.301
B0-20gon-2_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.663	6.278	-1.266
B0-20gon-3_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.590	5.358	-1.296
B0-20gon-3_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.456	5.147	-1.253
B0-20gon-3_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.790	6.504	-1.302
B0-20gon-3_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.656	6.293	-1.259
B0-8gon-1_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.597	5.360	-1.291
B0-8gon-1_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.464	5.149	-1.242
B0-8gon-1_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.791	6.495	-1.307
B0-8gon-1_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.658	6.285	-1.257
B0-8gon-2_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.595	5.350	-1.287
B0-8gon-2_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.461	5.139	-1.246
B0-8gon-2_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.793	6.492	-1.299
B0-8gon-2_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.658	6.280	-1.258
B0-8gon-3_xyz.seg	I-Beam-B	a	-4.480	5.163	-1.257	-4.597	5.355	-1.288
B0-8gon-3_xyz.seg	I-Beam-B	b	-4.612	5.374	-1.253	-4.464	5.143	-1.246
B0-8gon-3_xyz.seg	I-Beam-B	c	-2.666	6.289	-1.278	-2.790	6.490	-1.299
B0-8gon-3_xyz.seg	I-Beam-B	d	-2.800	6.501	-1.275	-2.657	6.278	-1.257
B30-12gon-1_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.094	4.692	-1.255
B30-12gon-1_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.317	4.810	-1.285
B30-12gon-1_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.098	6.579	-1.264
B30-12gon-1_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.321	6.697	-1.293

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B30-12gon-2_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.310	4.801	-1.285
B30-12gon-2_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.086	4.685	-1.261
B30-12gon-2_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.326	6.695	-1.286
B30-12gon-2_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.102	6.578	-1.261
B30-12gon-3_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.092	4.694	-1.249
B30-12gon-3_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.316	4.812	-1.275
B30-12gon-3_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.096	6.581	-1.267
B30-12gon-3_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.319	6.698	-1.293
B30-20gon-1_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.086	4.697	-1.253
B30-20gon-1_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.310	4.814	-1.280
B30-20gon-1_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.099	6.589	-1.270
B30-20gon-1_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.323	6.705	-1.298
B30-20gon-2_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.090	4.681	-1.259
B30-20gon-2_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.314	4.797	-1.287
B30-20gon-2_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.106	6.574	-1.265
B30-20gon-2_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.330	6.690	-1.293
B30-20gon-3_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.088	4.690	-1.260
B30-20gon-3_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.312	4.807	-1.288
B30-20gon-3_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.102	6.582	-1.266
B30-20gon-3_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.325	6.698	-1.295
B30-8gon-1_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.089	4.695	-1.253
B30-8gon-1_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.313	4.813	-1.279
B30-8gon-1_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.094	6.582	-1.262
B30-8gon-1_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.317	6.700	-1.288
B30-8gon-2_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.094	4.687	-1.252
B30-8gon-2_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.317	4.803	-1.285
B30-8gon-2_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.107	6.579	-1.258
B30-8gon-2_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.331	6.695	-1.292
B30-8gon-3_xyz.seg	I-Beam-B	a	-4.060	4.789	-1.261	-4.094	4.698	-1.252
B30-8gon-3_xyz.seg	I-Beam-B	b	-4.282	4.903	-1.252	-4.316	4.815	-1.285
B30-8gon-3_xyz.seg	I-Beam-B	c	-3.071	6.678	-1.277	-3.097	6.584	-1.262
B30-8gon-3_xyz.seg	I-Beam-B	d	-3.294	6.792	-1.269	-3.320	6.701	-1.295

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B45-12gon-1_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-3.809	4.531	-1.261
B45-12gon-1_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-4.057	4.585	-1.281
B45-12gon-1_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.348	6.614	-1.268
B45-12gon-1_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.596	6.669	-1.288
B45-12gon-2_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-3.808	4.527	-1.261
B45-12gon-2_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-4.055	4.581	-1.284
B45-12gon-2_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.354	6.612	-1.266
B45-12gon-2_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.601	6.666	-1.289
B45-12gon-3_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-4.057	4.580	-1.287
B45-12gon-3_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-3.810	4.526	-1.266
B45-12gon-3_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.599	6.664	-1.286
B45-12gon-3_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.352	6.610	-1.265
B45-20gon-1_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-4.052	4.567	-1.291
B45-20gon-1_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-3.805	4.513	-1.274
B45-20gon-1_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.600	6.652	-1.289
B45-20gon-1_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.353	6.598	-1.273
B45-20gon-2_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-3.810	4.519	-1.262
B45-20gon-2_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-4.057	4.574	-1.276
B45-20gon-2_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.349	6.603	-1.273
B45-20gon-2_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.596	6.657	-1.287
B45-20gon-3_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-4.067	4.567	-1.293
B45-20gon-3_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-3.820	4.511	-1.273
B45-20gon-3_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.595	6.648	-1.288
B45-20gon-3_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.348	6.592	-1.268
B45-8gon-1_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-3.816	4.531	-1.254
B45-8gon-1_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-4.062	4.586	-1.283
B45-8gon-1_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.350	6.613	-1.268
B45-8gon-1_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.596	6.668	-1.296
B45-8gon-2_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-3.810	4.529	-1.257
B45-8gon-2_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-4.057	4.583	-1.281
B45-8gon-2_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.352	6.613	-1.264
B45-8gon-2_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.598	6.667	-1.289

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B45-8gon-3_xyz.seg	I-Beam-B	a	-3.792	4.682	-1.263	-3.814	4.525	-1.255
B45-8gon-3_xyz.seg	I-Beam-B	b	-4.038	4.731	-1.254	-4.061	4.580	-1.280
B45-8gon-3_xyz.seg	I-Beam-B	c	-3.345	6.767	-1.275	-3.349	6.607	-1.266
B45-8gon-3_xyz.seg	I-Beam-B	d	-3.591	6.817	-1.266	-3.596	6.662	-1.291
B60-12gon-1_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.505	4.452	-1.269
B60-12gon-1_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.759	4.447	-1.282
B60-12gon-1_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.552	6.586	-1.274
B60-12gon-1_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.805	6.580	-1.288
B60-12gon-2_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.753	4.443	-1.290
B60-12gon-2_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.499	4.450	-1.274
B60-12gon-2_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.810	6.576	-1.285
B60-12gon-2_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.556	6.583	-1.270
B60-12gon-3_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.753	4.439	-1.290
B60-12gon-3_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.499	4.446	-1.276
B60-12gon-3_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.810	6.572	-1.283
B60-12gon-3_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.556	6.579	-1.269
B60-20gon-1_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.758	4.427	-1.302
B60-20gon-1_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.505	4.434	-1.279
B60-20gon-1_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.818	6.559	-1.283
B60-20gon-1_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.565	6.566	-1.261
B60-20gon-2_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.511	4.458	-1.264
B60-20gon-2_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.764	4.453	-1.288
B60-20gon-2_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.555	6.591	-1.275
B60-20gon-2_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.808	6.586	-1.299
B60-20gon-3_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.758	4.448	-1.298
B60-20gon-3_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.505	4.455	-1.275
B60-20gon-3_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.814	6.581	-1.293
B60-20gon-3_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.561	6.588	-1.270
B60-8gon-1_xyz.seg	I-Beam-A	a	-3.520	4.655	-1.269	-3.710	3.985	-1.262
B60-8gon-1_xyz.seg	I-Beam-A	b	-3.772	4.642	-1.259	-3.531	3.989	-1.248
B60-8gon-1_xyz.seg	I-Beam-A	c	-3.600	6.787	-1.272	-3.785	7.034	-1.258
B60-8gon-1_xyz.seg	I-Beam-A	d	-3.851	6.775	-1.261	-3.606	7.038	-1.243

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B60-8gon-2_xyz.seg	I-Beam-B	a	-3.520	4.655	-1.269	-3.508	4.449	-1.267
B60-8gon-2_xyz.seg	I-Beam-B	b	-3.772	4.642	-1.259	-3.761	4.444	-1.287
B60-8gon-2_xyz.seg	I-Beam-B	c	-3.600	6.787	-1.272	-3.556	6.582	-1.267
B60-8gon-2_xyz.seg	I-Beam-B	d	-3.851	6.775	-1.261	-3.809	6.577	-1.288
B60-8gon-3_xyz.seg	I-Beam-A	a	-3.520	4.655	-1.269	-3.708	3.985	-1.263
B60-8gon-3_xyz.seg	I-Beam-A	b	-3.772	4.642	-1.259	-3.529	3.990	-1.252
B60-8gon-3_xyz.seg	I-Beam-A	c	-3.600	6.787	-1.272	-3.785	7.034	-1.253
B60-8gon-3_xyz.seg	I-Beam-A	d	-3.851	6.775	-1.261	-3.606	7.039	-1.242
B90-12gon-1_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.711	4.362	-1.687
B90-12gon-1_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.929	4.232	-1.691
B90-12gon-1_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.807	6.191	-1.593
B90-12gon-1_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.024	6.060	-1.597
B90-12gon-2_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.716	4.368	-1.678
B90-12gon-2_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.935	4.240	-1.687
B90-12gon-2_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.795	6.207	-1.605
B90-12gon-2_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.014	6.079	-1.613
B90-12gon-3_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.712	4.356	-1.682
B90-12gon-3_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.931	4.227	-1.688
B90-12gon-3_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.796	6.191	-1.593
B90-12gon-3_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.015	6.062	-1.599
B90-20gon-1_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.677	4.341	-1.716
B90-20gon-1_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.893	4.209	-1.729
B90-20gon-1_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.793	6.154	-1.573
B90-20gon-1_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.010	6.021	-1.585
B90-20gon-2_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.724	4.404	-1.691
B90-20gon-2_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.942	4.273	-1.702
B90-20gon-2_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.826	6.228	-1.581
B90-20gon-2_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.043	6.097	-1.592
B90-20gon-3_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-4.589	3.322	-0.283
B90-20gon-3_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-4.737	3.116	-0.281
B90-20gon-3_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-5.147	3.697	-2.308
B90-20gon-3_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-5.295	3.491	-2.306

Table A.5. Continued.

File Name	I-Beam Selected	Site Point	SMS Coordinates (m)			Bounding Box Coordinates (m)		
			x	y	z	x	y	z
B90-8gon-1_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.729	4.411	-1.679
B90-8gon-1_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.942	4.274	-1.664
B90-8gon-1_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.881	6.206	-1.632
B90-8gon-1_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.094	6.069	-1.617
B90-8gon-2_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.731	4.378	-1.697
B90-8gon-2_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.949	4.247	-1.690
B90-8gon-2_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.829	6.205	-1.605
B90-8gon-2_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.046	6.074	-1.597
B90-8gon-3_xyz.seg	I-Beam-B	a	-2.986	4.823	-1.273	-2.725	4.387	-1.672
B90-8gon-3_xyz.seg	I-Beam-B	b	-3.198	4.689	-1.264	-2.943	4.256	-1.675
B90-8gon-3_xyz.seg	I-Beam-B	c	-4.102	6.640	-1.266	-3.819	6.217	-1.622
B90-8gon-3_xyz.seg	I-Beam-B	d	-4.315	6.505	-1.257	-4.037	6.087	-1.625

Table A.6. Errors between SMS coordinates of four flange points and the computed coordinates using the TIN segmentation process.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A0-12gon-1_xyz.seg	I-Beam-A	a	-0.031	0.033	0.005
A0-12gon-1_xyz.seg	I-Beam-A	b	-0.031	0.028	0.011
A0-12gon-1_xyz.seg	I-Beam-A	c	-0.026	0.026	-0.011
A0-12gon-1_xyz.seg	I-Beam-A	d	-0.026	0.025	-0.003
A0-12gon-2_xyz.seg	I-Beam-A	a	-0.035	0.032	0.008
A0-12gon-2_xyz.seg	I-Beam-A	b	-0.035	0.027	0.013
A0-12gon-2_xyz.seg	I-Beam-A	c	-0.029	0.023	-0.012
A0-12gon-2_xyz.seg	I-Beam-A	d	-0.029	0.022	-0.005
A0-12gon-3_xyz.seg	I-Beam-A	a	-0.027	0.036	0.005
A0-12gon-3_xyz.seg	I-Beam-A	b	-0.027	0.032	0.010
A0-12gon-3_xyz.seg	I-Beam-A	c	-0.022	0.029	-0.010
A0-12gon-3_xyz.seg	I-Beam-A	d	-0.021	0.027	-0.003
A0-20gon-1_xyz.seg	I-Beam-A	a	-0.016	0.040	0.010
A0-20gon-1_xyz.seg	I-Beam-A	b	-0.017	0.036	0.012
A0-20gon-1_xyz.seg	I-Beam-A	c	-0.016	0.040	-0.008
A0-20gon-1_xyz.seg	I-Beam-A	d	-0.016	0.039	-0.004
A0-20gon-2_xyz.seg	I-Beam-A	a	-0.030	0.031	0.011
A0-20gon-2_xyz.seg	I-Beam-A	b	-0.030	0.026	0.015
A0-20gon-2_xyz.seg	I-Beam-A	c	-0.027	0.027	-0.009
A0-20gon-2_xyz.seg	I-Beam-A	d	-0.026	0.025	-0.003
A0-20gon-3_xyz.seg	I-Beam-A	a	-0.016	0.037	0.009
A0-20gon-3_xyz.seg	I-Beam-A	b	-0.016	0.032	0.011
A0-20gon-3_xyz.seg	I-Beam-A	c	-0.017	0.040	-0.006
A0-20gon-3_xyz.seg	I-Beam-A	d	-0.017	0.038	-0.001
A0-8gon-1_xyz.seg	I-Beam-A	a	-0.024	0.038	0.006
A0-8gon-1_xyz.seg	I-Beam-A	b	-0.024	0.034	0.014
A0-8gon-1_xyz.seg	I-Beam-A	c	-0.020	0.032	-0.015
A0-8gon-1_xyz.seg	I-Beam-A	d	-0.019	0.030	-0.004

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A0-8gon-2_xyz.seg	I-Beam-A	a	-0.024	0.036	0.007
A0-8gon-2_xyz.seg	I-Beam-A	b	-0.024	0.032	0.016
A0-8gon-2_xyz.seg	I-Beam-A	c	-0.018	0.027	-0.015
A0-8gon-2_xyz.seg	I-Beam-A	d	-0.018	0.026	-0.004
A0-8gon-3_xyz.seg	I-Beam-A	a	-0.022	0.035	0.008
A0-8gon-3_xyz.seg	I-Beam-A	b	-0.022	0.031	0.015
A0-8gon-3_xyz.seg	I-Beam-A	c	-0.016	0.027	-0.016
A0-8gon-3_xyz.seg	I-Beam-A	d	-0.016	0.026	-0.007
A30-12gon-1_xyz.seg	I-Beam-A	a	0.058	0.176	0.005
A30-12gon-1_xyz.seg	I-Beam-A	b	0.061	0.175	0.013
A30-12gon-1_xyz.seg	I-Beam-A	c	0.063	0.174	-0.009
A30-12gon-1_xyz.seg	I-Beam-A	d	0.065	0.172	0.001
A30-12gon-2_xyz.seg	I-Beam-A	a	0.060	0.180	0.005
A30-12gon-2_xyz.seg	I-Beam-A	b	0.063	0.178	0.013
A30-12gon-2_xyz.seg	I-Beam-A	c	0.065	0.178	-0.010
A30-12gon-2_xyz.seg	I-Beam-A	d	0.067	0.175	0.000
A30-12gon-3_xyz.seg	I-Beam-A	a	0.058	0.174	0.003
A30-12gon-3_xyz.seg	I-Beam-A	b	0.061	0.173	0.011
A30-12gon-3_xyz.seg	I-Beam-A	c	0.061	0.173	-0.006
A30-12gon-3_xyz.seg	I-Beam-A	d	0.063	0.170	0.004
A30-20gon-1_xyz.seg	I-Beam-A	a	0.054	0.172	0.002
A30-20gon-1_xyz.seg	I-Beam-A	b	0.057	0.170	0.006
A30-20gon-1_xyz.seg	I-Beam-A	c	0.058	0.170	-0.005
A30-20gon-1_xyz.seg	I-Beam-A	d	0.060	0.167	0.001
A30-20gon-2_xyz.seg	I-Beam-A	a	0.062	0.190	0.009
A30-20gon-2_xyz.seg	I-Beam-A	b	0.065	0.189	0.018
A30-20gon-2_xyz.seg	I-Beam-A	c	0.072	0.186	-0.005
A30-20gon-2_xyz.seg	I-Beam-A	d	0.074	0.184	0.005
A30-20gon-3_xyz.seg	I-Beam-A	a	0.060	0.179	0.010
A30-20gon-3_xyz.seg	I-Beam-A	b	0.063	0.178	0.020
A30-20gon-3_xyz.seg	I-Beam-A	c	0.068	0.175	-0.007
A30-20gon-3_xyz.seg	I-Beam-A	d	0.070	0.173	0.004

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A30-8gon-1_xyz.seg	I-Beam-A	a	0.065	0.184	0.005
A30-8gon-1_xyz.seg	I-Beam-A	b	0.067	0.183	0.014
A30-8gon-1_xyz.seg	I-Beam-A	c	0.068	0.183	-0.013
A30-8gon-1_xyz.seg	I-Beam-A	d	0.070	0.180	-0.002
A30-8gon-2_xyz.seg	I-Beam-A	a	0.061	0.177	0.002
A30-8gon-2_xyz.seg	I-Beam-A	b	0.064	0.176	0.012
A30-8gon-2_xyz.seg	I-Beam-A	c	0.064	0.176	-0.010
A30-8gon-2_xyz.seg	I-Beam-A	d	0.066	0.173	0.002
A30-8gon-3_xyz.seg	I-Beam-A	a	0.061	0.175	0.003
A30-8gon-3_xyz.seg	I-Beam-A	b	0.064	0.173	0.012
A30-8gon-3_xyz.seg	I-Beam-A	c	0.063	0.174	-0.012
A30-8gon-3_xyz.seg	I-Beam-A	d	0.064	0.172	-0.002
A45-12gon-1_xyz.seg	I-Beam-A	a	0.039	0.275	0.008
A45-12gon-1_xyz.seg	I-Beam-A	b	0.040	0.273	0.017
A45-12gon-1_xyz.seg	I-Beam-A	c	0.033	0.275	-0.009
A45-12gon-1_xyz.seg	I-Beam-A	d	0.035	0.274	0.001
A45-12gon-2_xyz.seg	I-Beam-A	a	0.212	0.243	0.018
A45-12gon-2_xyz.seg	I-Beam-A	b	-0.137	0.317	0.012
A45-12gon-2_xyz.seg	I-Beam-A	c	0.212	0.242	-0.004
A45-12gon-2_xyz.seg	I-Beam-A	d	-0.136	0.318	-0.009
A45-12gon-3_xyz.seg	I-Beam-A	a	0.038	0.273	0.007
A45-12gon-3_xyz.seg	I-Beam-A	b	0.038	0.270	0.015
A45-12gon-3_xyz.seg	I-Beam-A	c	0.033	0.272	-0.011
A45-12gon-3_xyz.seg	I-Beam-A	d	0.036	0.271	-0.001
A45-20gon-1_xyz.seg	I-Beam-A	a	0.038	0.256	0.002
A45-20gon-1_xyz.seg	I-Beam-A	b	0.038	0.254	0.010
A45-20gon-1_xyz.seg	I-Beam-A	c	0.028	0.257	-0.003
A45-20gon-1_xyz.seg	I-Beam-A	d	0.030	0.256	0.006
A45-20gon-2_xyz.seg	I-Beam-A	a	0.212	0.245	0.020
A45-20gon-2_xyz.seg	I-Beam-A	b	-0.138	0.320	0.014
A45-20gon-2_xyz.seg	I-Beam-A	c	0.215	0.244	0.000
A45-20gon-2_xyz.seg	I-Beam-A	d	-0.133	0.319	-0.005

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A45-20gon-3_xyz.seg	I-Beam-A	a	0.046	0.281	0.008
A45-20gon-3_xyz.seg	I-Beam-A	b	0.046	0.278	0.021
A45-20gon-3_xyz.seg	I-Beam-A	c	0.037	0.282	-0.004
A45-20gon-3_xyz.seg	I-Beam-A	d	0.039	0.280	0.010
A45-8gon-1_xyz.seg	I-Beam-A	a	0.040	0.273	0.005
A45-8gon-1_xyz.seg	I-Beam-A	b	0.040	0.271	0.015
A45-8gon-1_xyz.seg	I-Beam-A	c	0.033	0.274	-0.012
A45-8gon-1_xyz.seg	I-Beam-A	d	0.035	0.273	-0.001
A45-8gon-2_xyz.seg	I-Beam-A	a	0.215	0.238	0.015
A45-8gon-2_xyz.seg	I-Beam-A	b	-0.135	0.313	0.008
A45-8gon-2_xyz.seg	I-Beam-A	c	0.208	0.238	-0.005
A45-8gon-2_xyz.seg	I-Beam-A	d	-0.140	0.314	-0.011
A45-8gon-3_xyz.seg	I-Beam-A	a	0.040	0.272	0.004
A45-8gon-3_xyz.seg	I-Beam-A	b	0.040	0.269	0.014
A45-8gon-3_xyz.seg	I-Beam-A	c	0.032	0.273	-0.011
A45-8gon-3_xyz.seg	I-Beam-A	d	0.034	0.271	-0.001
A60-12gon-1_xyz.seg	I-Beam-A	a	0.148	0.346	0.014
A60-12gon-1_xyz.seg	I-Beam-A	b	-0.207	0.332	0.009
A60-12gon-1_xyz.seg	I-Beam-A	c	0.135	0.345	0.003
A60-12gon-1_xyz.seg	I-Beam-A	d	-0.221	0.332	0.001
A60-12gon-2_xyz.seg	I-Beam-A	a	0.148	0.351	0.015
A60-12gon-2_xyz.seg	I-Beam-A	b	-0.207	0.337	0.011
A60-12gon-2_xyz.seg	I-Beam-A	c	0.133	0.350	0.005
A60-12gon-2_xyz.seg	I-Beam-A	d	-0.223	0.337	0.004
A60-12gon-3_xyz.seg	I-Beam-A	a	0.149	0.345	0.015
A60-12gon-3_xyz.seg	I-Beam-A	b	-0.207	0.331	0.009
A60-12gon-3_xyz.seg	I-Beam-A	c	0.137	0.344	0.005
A60-12gon-3_xyz.seg	I-Beam-A	d	-0.219	0.331	0.001
A60-20gon-1_xyz.seg	I-Beam-A	a	0.145	0.367	0.019
A60-20gon-1_xyz.seg	I-Beam-A	b	-0.211	0.352	0.013
A60-20gon-1_xyz.seg	I-Beam-A	c	0.140	0.366	0.006
A60-20gon-1_xyz.seg	I-Beam-A	d	-0.216	0.353	0.003

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A60-20gon-2_xyz.seg	I-Beam-A	a	0.143	0.360	0.017
A60-20gon-2_xyz.seg	I-Beam-A	b	-0.213	0.346	0.014
A60-20gon-2_xyz.seg	I-Beam-A	c	0.137	0.360	0.004
A60-20gon-2_xyz.seg	I-Beam-A	d	-0.219	0.346	0.003
A60-20gon-3_xyz.seg	I-Beam-A	a	0.145	0.368	0.020
A60-20gon-3_xyz.seg	I-Beam-A	b	-0.210	0.354	0.014
A60-20gon-3_xyz.seg	I-Beam-A	c	0.140	0.368	0.004
A60-20gon-3_xyz.seg	I-Beam-A	d	-0.216	0.354	0.000
A60-8gon-1_xyz.seg	I-Beam-A	a	0.147	0.364	0.018
A60-8gon-1_xyz.seg	I-Beam-A	b	-0.209	0.349	0.011
A60-8gon-1_xyz.seg	I-Beam-A	c	0.138	0.363	0.002
A60-8gon-1_xyz.seg	I-Beam-A	d	-0.218	0.350	-0.002
A60-8gon-2_xyz.seg	I-Beam-A	a	0.146	0.359	0.015
A60-8gon-2_xyz.seg	I-Beam-A	b	-0.210	0.345	0.009
A60-8gon-2_xyz.seg	I-Beam-A	c	0.137	0.359	-0.001
A60-8gon-2_xyz.seg	I-Beam-A	d	-0.219	0.346	-0.004
A60-8gon-3_xyz.seg	I-Beam-A	a	0.146	0.367	0.016
A60-8gon-3_xyz.seg	I-Beam-A	b	-0.210	0.353	0.011
A60-8gon-3_xyz.seg	I-Beam-A	c	0.137	0.366	-0.002
A60-8gon-3_xyz.seg	I-Beam-A	d	-0.219	0.353	-0.005
A90-12gon-1_xyz.seg	I-Beam-B	a	-0.297	0.536	0.549
A90-12gon-1_xyz.seg	I-Beam-B	b	-0.224	0.566	0.541
A90-12gon-1_xyz.seg	I-Beam-B	c	-0.872	1.272	0.341
A90-12gon-1_xyz.seg	I-Beam-B	d	-0.802	1.303	0.336
A90-12gon-2_xyz.seg	I-Beam-B	a	-1.861	-2.670	0.145
A90-12gon-2_xyz.seg	I-Beam-B	b	-1.773	-2.801	0.059
A90-12gon-2_xyz.seg	I-Beam-B	c	-3.377	1.966	-0.445
A90-12gon-2_xyz.seg	I-Beam-B	d	-3.292	1.836	-0.529
A90-12gon-3_xyz.seg	I-Beam-B	a	-0.333	0.446	0.504
A90-12gon-3_xyz.seg	I-Beam-B	b	-0.264	0.484	0.512
A90-12gon-3_xyz.seg	I-Beam-B	c	-0.823	1.226	0.344
A90-12gon-3_xyz.seg	I-Beam-B	d	-0.757	1.266	0.353

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
A90-20gon-1_xyz.seg	I-Beam-B	a	-0.266	0.408	0.473
A90-20gon-1_xyz.seg	I-Beam-B	b	-0.194	0.440	0.479
A90-20gon-1_xyz.seg	I-Beam-B	c	-0.812	1.151	0.372
A90-20gon-1_xyz.seg	I-Beam-B	d	-0.742	1.184	0.379
A90-20gon-2_xyz.seg	I-Beam-A	a	-0.420	0.711	0.508
A90-20gon-2_xyz.seg	I-Beam-A	b	-0.413	0.707	0.514
A90-20gon-2_xyz.seg	I-Beam-A	c	-0.484	0.674	0.378
A90-20gon-2_xyz.seg	I-Beam-A	d	-0.480	0.672	0.387
A90-20gon-3_xyz.seg	I-Beam-B	a	-0.271	0.353	0.487
A90-20gon-3_xyz.seg	I-Beam-B	b	-0.202	0.391	0.497
A90-20gon-3_xyz.seg	I-Beam-B	c	-0.768	1.131	0.318
A90-20gon-3_xyz.seg	I-Beam-B	d	-0.702	1.169	0.331
A90-8gon-1_xyz.seg	I-Beam-B	a	1.748	-3.253	0.945
A90-8gon-1_xyz.seg	I-Beam-B	b	1.550	-3.101	0.907
A90-8gon-1_xyz.seg	I-Beam-B	c	0.195	-0.988	-1.154
A90-8gon-1_xyz.seg	I-Beam-B	d	-0.006	-0.835	-1.190
A90-8gon-2_xyz.seg	I-Beam-B	a	1.778	-3.567	-1.133
A90-8gon-2_xyz.seg	I-Beam-B	b	1.578	-3.416	-1.165
A90-8gon-2_xyz.seg	I-Beam-B	c	0.174	-0.690	0.990
A90-8gon-2_xyz.seg	I-Beam-B	d	-0.030	-0.537	0.960
A90-8gon-3_xyz.seg	I-Beam-B	a	-1.618	-2.774	-0.065
A90-8gon-3_xyz.seg	I-Beam-B	b	-2.002	-2.844	0.028
A90-8gon-3_xyz.seg	I-Beam-B	c	-3.146	1.918	-0.409
A90-8gon-3_xyz.seg	I-Beam-B	d	-3.533	1.849	-0.314
B0-12gon-1_xyz.seg	I-Beam-B	a	0.115	-0.188	0.027
B0-12gon-1_xyz.seg	I-Beam-B	b	-0.152	0.235	-0.003
B0-12gon-1_xyz.seg	I-Beam-B	c	0.125	-0.202	0.017
B0-12gon-1_xyz.seg	I-Beam-B	d	-0.144	0.223	-0.014
B0-12gon-2_xyz.seg	I-Beam-B	a	0.122	-0.183	0.028
B0-12gon-2_xyz.seg	I-Beam-B	b	-0.145	0.240	-0.002
B0-12gon-2_xyz.seg	I-Beam-B	c	0.132	-0.198	0.019
B0-12gon-2_xyz.seg	I-Beam-B	d	-0.136	0.227	-0.012

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B0-12gon-3_xyz.seg	I-Beam-B	a	0.120	-0.194	0.031
B0-12gon-3_xyz.seg	I-Beam-B	b	-0.146	0.228	-0.007
B0-12gon-3_xyz.seg	I-Beam-B	c	0.126	-0.203	0.025
B0-12gon-3_xyz.seg	I-Beam-B	d	-0.141	0.220	-0.014
B0-20gon-1_xyz.seg	I-Beam-B	a	0.112	-0.196	0.039
B0-20gon-1_xyz.seg	I-Beam-B	b	-0.154	0.225	0.000
B0-20gon-1_xyz.seg	I-Beam-B	c	0.125	-0.215	0.022
B0-20gon-1_xyz.seg	I-Beam-B	d	-0.143	0.208	-0.018
B0-20gon-2_xyz.seg	I-Beam-B	a	0.120	-0.185	0.034
B0-20gon-2_xyz.seg	I-Beam-B	b	-0.147	0.238	0.003
B0-20gon-2_xyz.seg	I-Beam-B	c	0.131	-0.201	0.023
B0-20gon-2_xyz.seg	I-Beam-B	d	-0.137	0.223	-0.009
B0-20gon-3_xyz.seg	I-Beam-B	a	0.110	-0.195	0.039
B0-20gon-3_xyz.seg	I-Beam-B	b	-0.157	0.227	0.000
B0-20gon-3_xyz.seg	I-Beam-B	c	0.124	-0.215	0.024
B0-20gon-3_xyz.seg	I-Beam-B	d	-0.144	0.208	-0.016
B0-8gon-1_xyz.seg	I-Beam-B	a	0.117	-0.196	0.034
B0-8gon-1_xyz.seg	I-Beam-B	b	-0.148	0.225	-0.011
B0-8gon-1_xyz.seg	I-Beam-B	c	0.125	-0.206	0.028
B0-8gon-1_xyz.seg	I-Beam-B	d	-0.142	0.216	-0.018
B0-8gon-2_xyz.seg	I-Beam-B	a	0.115	-0.187	0.030
B0-8gon-2_xyz.seg	I-Beam-B	b	-0.152	0.235	-0.006
B0-8gon-2_xyz.seg	I-Beam-B	c	0.126	-0.203	0.020
B0-8gon-2_xyz.seg	I-Beam-B	d	-0.142	0.220	-0.017
B0-8gon-3_xyz.seg	I-Beam-B	a	0.117	-0.192	0.031
B0-8gon-3_xyz.seg	I-Beam-B	b	-0.149	0.230	-0.006
B0-8gon-3_xyz.seg	I-Beam-B	c	0.124	-0.201	0.021
B0-8gon-3_xyz.seg	I-Beam-B	d	-0.143	0.223	-0.018
B30-12gon-1_xyz.seg	I-Beam-B	a	0.034	0.096	-0.006
B30-12gon-1_xyz.seg	I-Beam-B	b	0.035	0.093	0.033
B30-12gon-1_xyz.seg	I-Beam-B	c	0.028	0.099	-0.014
B30-12gon-1_xyz.seg	I-Beam-B	d	0.028	0.095	0.024

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B30-12gon-2_xyz.seg	I-Beam-B	a	0.250	-0.013	0.024
B30-12gon-2_xyz.seg	I-Beam-B	b	-0.197	0.218	0.008
B30-12gon-2_xyz.seg	I-Beam-B	c	0.256	-0.017	0.008
B30-12gon-2_xyz.seg	I-Beam-B	d	-0.192	0.214	-0.008
B30-12gon-3_xyz.seg	I-Beam-B	a	0.032	0.095	-0.012
B30-12gon-3_xyz.seg	I-Beam-B	b	0.033	0.091	0.023
B30-12gon-3_xyz.seg	I-Beam-B	c	0.025	0.097	-0.011
B30-12gon-3_xyz.seg	I-Beam-B	d	0.026	0.094	0.024
B30-20gon-1_xyz.seg	I-Beam-B	a	0.027	0.091	-0.009
B30-20gon-1_xyz.seg	I-Beam-B	b	0.028	0.089	0.027
B30-20gon-1_xyz.seg	I-Beam-B	c	0.029	0.089	-0.007
B30-20gon-1_xyz.seg	I-Beam-B	d	0.030	0.087	0.029
B30-20gon-2_xyz.seg	I-Beam-B	a	0.030	0.108	-0.002
B30-20gon-2_xyz.seg	I-Beam-B	b	0.032	0.106	0.035
B30-20gon-2_xyz.seg	I-Beam-B	c	0.036	0.104	-0.012
B30-20gon-2_xyz.seg	I-Beam-B	d	0.037	0.102	0.024
B30-20gon-3_xyz.seg	I-Beam-B	a	0.029	0.099	-0.001
B30-20gon-3_xyz.seg	I-Beam-B	b	0.030	0.096	0.036
B30-20gon-3_xyz.seg	I-Beam-B	c	0.031	0.096	-0.011
B30-20gon-3_xyz.seg	I-Beam-B	d	0.032	0.094	0.026
B30-8gon-1_xyz.seg	I-Beam-B	a	0.030	0.094	-0.008
B30-8gon-1_xyz.seg	I-Beam-B	b	0.030	0.090	0.027
B30-8gon-1_xyz.seg	I-Beam-B	c	0.023	0.096	-0.016
B30-8gon-1_xyz.seg	I-Beam-B	d	0.024	0.092	0.019
B30-8gon-2_xyz.seg	I-Beam-B	a	0.034	0.102	-0.009
B30-8gon-2_xyz.seg	I-Beam-B	b	0.035	0.099	0.033
B30-8gon-2_xyz.seg	I-Beam-B	c	0.037	0.099	-0.019
B30-8gon-2_xyz.seg	I-Beam-B	d	0.037	0.097	0.023
B30-8gon-3_xyz.seg	I-Beam-B	a	0.034	0.091	-0.009
B30-8gon-3_xyz.seg	I-Beam-B	b	0.034	0.088	0.033
B30-8gon-3_xyz.seg	I-Beam-B	c	0.026	0.094	-0.015
B30-8gon-3_xyz.seg	I-Beam-B	d	0.026	0.091	0.026

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B45-12gon-1_xyz.seg	I-Beam-B	a	0.018	0.151	-0.002
B45-12gon-1_xyz.seg	I-Beam-B	b	0.019	0.145	0.027
B45-12gon-1_xyz.seg	I-Beam-B	c	0.003	0.153	-0.007
B45-12gon-1_xyz.seg	I-Beam-B	d	0.004	0.148	0.022
B45-12gon-2_xyz.seg	I-Beam-B	a	0.016	0.155	-0.003
B45-12gon-2_xyz.seg	I-Beam-B	b	0.018	0.150	0.029
B45-12gon-2_xyz.seg	I-Beam-B	c	0.009	0.155	-0.009
B45-12gon-2_xyz.seg	I-Beam-B	d	0.010	0.151	0.023
B45-12gon-3_xyz.seg	I-Beam-B	a	0.266	0.102	0.024
B45-12gon-3_xyz.seg	I-Beam-B	b	-0.227	0.205	0.012
B45-12gon-3_xyz.seg	I-Beam-B	c	0.254	0.103	0.011
B45-12gon-3_xyz.seg	I-Beam-B	d	-0.239	0.208	-0.002
B45-20gon-1_xyz.seg	I-Beam-B	a	0.261	0.115	0.027
B45-20gon-1_xyz.seg	I-Beam-B	b	-0.233	0.218	0.020
B45-20gon-1_xyz.seg	I-Beam-B	c	0.255	0.115	0.014
B45-20gon-1_xyz.seg	I-Beam-B	d	-0.238	0.219	0.007
B45-20gon-2_xyz.seg	I-Beam-B	a	0.018	0.162	-0.001
B45-20gon-2_xyz.seg	I-Beam-B	b	0.020	0.156	0.022
B45-20gon-2_xyz.seg	I-Beam-B	c	0.004	0.164	-0.002
B45-20gon-2_xyz.seg	I-Beam-B	d	0.005	0.160	0.021
B45-20gon-3_xyz.seg	I-Beam-B	a	0.276	0.115	0.030
B45-20gon-3_xyz.seg	I-Beam-B	b	-0.217	0.220	0.019
B45-20gon-3_xyz.seg	I-Beam-B	c	0.250	0.119	0.013
B45-20gon-3_xyz.seg	I-Beam-B	d	-0.243	0.225	0.002
B45-8gon-1_xyz.seg	I-Beam-B	a	0.024	0.151	-0.010
B45-8gon-1_xyz.seg	I-Beam-B	b	0.025	0.145	0.028
B45-8gon-1_xyz.seg	I-Beam-B	c	0.005	0.154	-0.008
B45-8gon-1_xyz.seg	I-Beam-B	d	0.005	0.149	0.030
B45-8gon-2_xyz.seg	I-Beam-B	a	0.019	0.153	-0.007
B45-8gon-2_xyz.seg	I-Beam-B	b	0.020	0.147	0.027
B45-8gon-2_xyz.seg	I-Beam-B	c	0.006	0.154	-0.011
B45-8gon-2_xyz.seg	I-Beam-B	d	0.007	0.150	0.022

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B45-8gon-3_xyz.seg	I-Beam-B	a	0.023	0.157	-0.008
B45-8gon-3_xyz.seg	I-Beam-B	b	0.024	0.150	0.026
B45-8gon-3_xyz.seg	I-Beam-B	c	0.004	0.159	-0.009
B45-8gon-3_xyz.seg	I-Beam-B	d	0.005	0.155	0.025
B60-12gon-1_xyz.seg	I-Beam-B	a	-0.014	0.203	-0.001
B60-12gon-1_xyz.seg	I-Beam-B	b	-0.013	0.195	0.023
B60-12gon-1_xyz.seg	I-Beam-B	c	-0.048	0.201	0.002
B60-12gon-1_xyz.seg	I-Beam-B	d	-0.046	0.195	0.026
B60-12gon-2_xyz.seg	I-Beam-B	a	0.233	0.212	0.021
B60-12gon-2_xyz.seg	I-Beam-B	b	-0.273	0.192	0.015
B60-12gon-2_xyz.seg	I-Beam-B	c	0.210	0.211	0.013
B60-12gon-2_xyz.seg	I-Beam-B	d	-0.295	0.192	0.009
B60-12gon-3_xyz.seg	I-Beam-B	a	0.233	0.216	0.021
B60-12gon-3_xyz.seg	I-Beam-B	b	-0.273	0.196	0.017
B60-12gon-3_xyz.seg	I-Beam-B	c	0.210	0.214	0.011
B60-12gon-3_xyz.seg	I-Beam-B	d	-0.295	0.196	0.008
B60-20gon-1_xyz.seg	I-Beam-B	a	0.239	0.229	0.032
B60-20gon-1_xyz.seg	I-Beam-B	b	-0.267	0.208	0.020
B60-20gon-1_xyz.seg	I-Beam-B	c	0.218	0.227	0.012
B60-20gon-1_xyz.seg	I-Beam-B	d	-0.286	0.209	-0.001
B60-20gon-2_xyz.seg	I-Beam-B	a	-0.009	0.197	-0.005
B60-20gon-2_xyz.seg	I-Beam-B	b	-0.008	0.189	0.028
B60-20gon-2_xyz.seg	I-Beam-B	c	-0.045	0.195	0.004
B60-20gon-2_xyz.seg	I-Beam-B	d	-0.043	0.189	0.038
B60-20gon-3_xyz.seg	I-Beam-B	a	0.238	0.207	0.029
B60-20gon-3_xyz.seg	I-Beam-B	b	-0.267	0.187	0.016
B60-20gon-3_xyz.seg	I-Beam-B	c	0.214	0.206	0.021
B60-20gon-3_xyz.seg	I-Beam-B	d	-0.290	0.188	0.009
B60-8gon-1_xyz.seg	I-Beam-A	a	0.190	0.670	-0.007
B60-8gon-1_xyz.seg	I-Beam-A	b	-0.241	0.653	-0.011
B60-8gon-1_xyz.seg	I-Beam-A	c	0.185	-0.247	-0.014
B60-8gon-1_xyz.seg	I-Beam-A	d	-0.245	-0.263	-0.018

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B60-8gon-2_xyz.seg	I-Beam-B	a	-0.012	0.206	-0.002
B60-8gon-2_xyz.seg	I-Beam-B	b	-0.011	0.198	0.028
B60-8gon-2_xyz.seg	I-Beam-B	c	-0.044	0.204	-0.004
B60-8gon-2_xyz.seg	I-Beam-B	d	-0.042	0.198	0.027
B60-8gon-3_xyz.seg	I-Beam-A	a	0.189	0.670	-0.006
B60-8gon-3_xyz.seg	I-Beam-A	b	-0.243	0.652	-0.007
B60-8gon-3_xyz.seg	I-Beam-A	c	0.185	-0.247	-0.018
B60-8gon-3_xyz.seg	I-Beam-A	d	-0.245	-0.263	-0.019
B90-12gon-1_xyz.seg	I-Beam-B	a	-0.274	0.461	0.414
B90-12gon-1_xyz.seg	I-Beam-B	b	-0.269	0.457	0.427
B90-12gon-1_xyz.seg	I-Beam-B	c	-0.295	0.449	0.327
B90-12gon-1_xyz.seg	I-Beam-B	d	-0.291	0.445	0.340
B90-12gon-2_xyz.seg	I-Beam-B	a	-0.270	0.455	0.405
B90-12gon-2_xyz.seg	I-Beam-B	b	-0.263	0.449	0.423
B90-12gon-2_xyz.seg	I-Beam-B	c	-0.307	0.433	0.338
B90-12gon-2_xyz.seg	I-Beam-B	d	-0.301	0.426	0.357
B90-12gon-3_xyz.seg	I-Beam-B	a	-0.274	0.468	0.409
B90-12gon-3_xyz.seg	I-Beam-B	b	-0.267	0.463	0.424
B90-12gon-3_xyz.seg	I-Beam-B	c	-0.306	0.449	0.327
B90-12gon-3_xyz.seg	I-Beam-B	d	-0.300	0.443	0.342
B90-20gon-1_xyz.seg	I-Beam-B	a	-0.309	0.482	0.443
B90-20gon-1_xyz.seg	I-Beam-B	b	-0.305	0.480	0.465
B90-20gon-1_xyz.seg	I-Beam-B	c	-0.309	0.486	0.306
B90-20gon-1_xyz.seg	I-Beam-B	d	-0.305	0.484	0.329
B90-20gon-2_xyz.seg	I-Beam-B	a	-0.261	0.419	0.418
B90-20gon-2_xyz.seg	I-Beam-B	b	-0.256	0.416	0.438
B90-20gon-2_xyz.seg	I-Beam-B	c	-0.276	0.412	0.315
B90-20gon-2_xyz.seg	I-Beam-B	d	-0.272	0.408	0.335
B90-20gon-3_xyz.seg	I-Beam-B	a	1.603	1.501	-0.990
B90-20gon-3_xyz.seg	I-Beam-B	b	1.539	1.573	-0.983
B90-20gon-3_xyz.seg	I-Beam-B	c	1.045	2.943	1.042
B90-20gon-3_xyz.seg	I-Beam-B	d	0.980	3.015	1.049

Table A.6. Continued.

File Name	I-Beam Selected	Site Point	Error (m)		
			dx	dy	dz
B90-8gon-1_xyz.seg	I-Beam-B	a	-0.257	0.412	0.406
B90-8gon-1_xyz.seg	I-Beam-B	b	-0.255	0.416	0.401
B90-8gon-1_xyz.seg	I-Beam-B	c	-0.221	0.434	0.365
B90-8gon-1_xyz.seg	I-Beam-B	d	-0.221	0.437	0.360
B90-8gon-2_xyz.seg	I-Beam-B	a	-0.254	0.445	0.424
B90-8gon-2_xyz.seg	I-Beam-B	b	-0.249	0.442	0.426
B90-8gon-2_xyz.seg	I-Beam-B	c	-0.273	0.435	0.338
B90-8gon-2_xyz.seg	I-Beam-B	d	-0.269	0.431	0.341
B90-8gon-3_xyz.seg	I-Beam-B	a	-0.261	0.437	0.399
B90-8gon-3_xyz.seg	I-Beam-B	b	-0.255	0.433	0.411
B90-8gon-3_xyz.seg	I-Beam-B	c	-0.283	0.423	0.355
B90-8gon-3_xyz.seg	I-Beam-B	d	-0.278	0.418	0.368