USE OF LASER RANGE FINDERS AND RANGE IMAGE ANALYSIS IN AUTOMATED ASSEMBLY TASKS

By

Nicolas Alvertos, Principal Investigator

and

Ivan D'Cunha, Graduate Research Assistant

Final Report
For the period ended August 15, 1990

Prepared for
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665

Under
Master Contract NAS1-18584
Task Authorization No. 70
Plesent W. Goode IV, Technical Monitor
ISD-Automation Technology Branch

Submitted by the
Old Dominion University Research Foundation
P.O. Box 6369
Norfolk, Virginia 23508-0369

September 1990
Use of Laser Range finders and Range Image Analysis in Automated Assembly Tasks

by

Nicolas Alvertos * and Ivan D' Cunha **

Abstract

In this research it has been proposed to study the effect of filtering processes on range images and also to evaluate the performance of two different laser range mappers. Median filtering had been utilized to remove noise from the range images. First and second order derivatives are then utilized to locate the similarities and dissimilarities between the processed and the original images. Range depth information is converted into spatial coordinates, and a set of coefficients which describe three dimensional objects is generated using the algorithm developed in the second phase of this research. Range images of spheres and cylinders are used for experimental purposes. An algorithm was also developed to compare the performance of two different laser range mappers based upon the range depth information of surfaces generated by each of the mappers. Further more, an approach based on two-dimensional analytic geometry is also proposed which serves as a basis for the recognition of regular three dimensional geometric objects.

* Assistant Professor, Department of Electrical and Computer Engineering, Old Dominion University, Norfolk, Virginia 23529-0246.
** Graduate Research Assistant, Department of Electrical and Computer Engineering, Old Dominion University, Norfolk, Virginia 23529-0246.
1. Introduction

The problem of 3-D object recognition has been an interesting research area for the past few years with tremendous scope of improvisations in every department of the recognition scheme. Unlike the recognition procedures developed for intensity based image information, the recent upsurge of several active and passive sensors extracting quality range information has lead to the involvement of explicit geometric shapes of the objects for the recognition schemes.

Range images share the same format of the intensity images (i.e. either of these images are two dimensional array of numbers), the only difference being that the numbers in the range images represent the distances between a sensor focal plane to points in space. The laser range finder is the most widely used sensor these days. The laser range finder makes use of a laser beam which scans the surfaces in the scene of observation from left to right and top to bottom. The distances thus obtained are measures of both depth and scanning angle. Until unless a specific algorithm demands a special form of these range images, for most of the time it is mainly the depth information which is utilized for the recognition process.

The range data obtained from a laser radar vision system is chiefly affected with two types of problems. The first called the Doppler shift, erupts essentially due to the way a laser radar system functions. Recently new radar vision systems have come in the market with an inbuilt doppler shift corrector which removes the distortions from the range data. The second problem, which is noise in the data picture (mainly salt and pepper) is generated on account of the improper wiring circuitry of the whole system.
The process by which doppler shift is corrected for our system is discussed in [1]. In this report we will be discussing about the median filter which to a large extent helps in filtering the noisy range data.

Median filtering was first suggested by Tukey [5] and since then has been widely adopted for two-dimensional image noise smoothing. The most distinguishing property of the median filter is that it preserves monotonic step edges, i.e., it does not blur sharp edges as most of the linear filters would do.

Range data from regular objects like spheres, cylinders and cones have been considered in this research and the effect of median filtering on each of these has been studied. A scheme to evaluate range data obtained from two different laser range mappers is also discussed. As the prime objective of this research is to come up with a automatic 3-D object classifier, a new approach based upon analytic geometry has been proposed for the recognition scheme.

2. Theoretical Development

Median Filtering

Conventionally, a rectangular window of size M x N is used in two dimensional median filtering. As in our case, experiments were carried out with square windows of mask sizes 3 x 3 and 5 x 5. As according to the common belief of the existence of salt and pepper at the edges, noise in the range images experimented in this research were some what distributed uniformly throughout. Irrespective of the mask size, the range information at every pixel in the image is replaced by the median of the the pixels contained in the M x M window centered at that point. Referring to figure 1, keeping in mind that the dark pixels correspond to the object and the white pixels to the
background, specks of white pixels inside the object refers to the salt noise and the specks of black pixels in the white background refers to the pepper noise. Figure 3 is obtained as a result of a 3 x 3 mask being moved over the entire image. The picture looks as sharp as the original image though some of the noise still exists. A 5 x 5 mask completely removes all the salt and pepper noise, but the image as seen in figure 4, to some extent has a low contrast, but at the same time has become more smoother than the original image.

Once a range image is filtered using a median filter of different masks, the next concern is to study the changes which have been brought about by filtering to the original data. Evaluating curvatures is one good way of distinguishing similarities and dissimilarities among the filtered images and the original range data.

First and second order derivatives are evaluated along the x- and y-axis to check the uniformity of the original and the filtered images. The first order derivative for a pixel \( A_{i,j} \) centered at \( i,j \) is given as:

\[
\frac{\partial A}{\partial x} = \frac{1}{2\varepsilon}[(A_{i+1,j+1} - A_{i,j+1}) + (A_{i+1,j} - A_{i,j})],
\]

and

\[
\frac{\partial A}{\partial y} = \frac{1}{2\varepsilon}[(A_{i+1,j+1} - A_{i,j+1}) + (A_{i,j+1} - A_{i,j})]
\]

Similarly the second order derivatives for a pixel centered at \( A_{i,j} \) is given as:

\[
\frac{\partial^2 A}{\partial x^2} = \frac{1}{\varepsilon^2}[A_{i-1,j} - 2E_{i,j} + E_{i+1,j}],
\]

and

\[
\frac{\partial^2 A}{\partial y^2} = \frac{1}{\varepsilon^2}[A_{i,j-1} - 2E_{i,j} + E_{i,j+1}],
\]

\( \varepsilon \) above refers to the spacing between picture cell centers.
A sign map whereupon relationship among two neighboring pixels with respect to the depth value, is also generated to make sure that the median filtering does not alter the original data to a large extent.

A second degree general quadric surface as we know is given by the relation,

\[ F(x,y,z) = ax^2 + by^2 + cz^2 + 2fyz + 2gzx + 2hxy + 2px + 2qy + 2rz + d = 0 \]

Using the approach formulated by Groshong and Bilbro [1,2] the ten coefficients, \(a, b, c, d, f, g, h, p, q, \) and \(r\) that uniquely describe a quadric surface are determined. Coefficients are obtained for each of the filtered images and their relationship with the coefficients evaluated for the original range data (one with the noise) are studied for each of the surfaces individually.

**Evaluation of the performance of two different laser range mappers.**

In the second phase of our research [1], an approach has been put forward for determining the performance of two different laser range mappers using a particular test object, i.e., depth maps are obtained for the same object using two different range mappers. In this report we have come up with an approach which evaluates the performance of two different range mappers based upon the depth information obtained for two different sizes of the same object, i.e., the test object had the same shape but is of different size. The object under consideration is a sphere.

**Theory**

Consider the general equation of the sphere which is in the form of

\[ (x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2 = 0 \]  \( (1) \)

where \(x_1, y_1, \) and \(z_1\) are the coordinates of the center of the sphere. Equation (1) can
also be expressed as

\[ x^2 + y^2 + z^2 + 2fx + 2gf + 2hz + d = 0 \quad (2) \]

It is to be noted that the coefficients of \( x^2, y^2, \) and \( z^2 \) are all equal to 1.

From analytic geometry we know that \( x_1, y_1, \) and \( z_1 \) from equation (1) are related to the coefficients of \( x, y, \) and \( z \) in equation (2) with the following relations:

\[
\begin{align*}
x_1 &= -2f \\
x_2 &= -2g \\
x_3 &= -2h
\end{align*}
\]

Once the coefficients \( f, g, \) and \( h \) are evaluated using the algorithm formulated by Groshong and Bilbro [2], the center of the sphere, i.e., \( x_1, y_1, \) and \( z_1 \) is evaluated using the above relationships. It is to be noted that the coefficients \( f, g, \) and \( h \) and the center of the sphere \( (x_1, y_1, z_1) \) evaluated experimentally, certainly do not denote the correct coefficients and the center respectively, since a small surface patch of the range data has been utilized to determine these coefficients.

For each set of the sphere range data generated using two different laser range mappers, the coordinates of the center of sphere is determined. A least square approach as discussed below is next utilized to comment upon the performance of each of these laser range mappers.

Let \( N \) be the total number of points (pixels) used to determine the coefficients of the sphere generated using laser system 1.
Then

\[ D_1 = \sum_{i=0}^{N} (x_i - x_1)^2 + (y_i - y_1)^2 + (z_i - z_1)^2 \]

where \( x_i, y_i, \) and \( z_i \) are the cartesian coordinates of each of the \( N \) depth points, and \( x_1, y_1, \) and \( z_1 \) refer to the center of the sphere.

Now

\[ \frac{\sqrt{D_1}}{N} \]

denotes the mean square error for the system 1.

A similar approach is carried over for the sphere data generated using system 2 and a mean square error is evaluated. The value of the mean square error determines which set of data is more closer to the data generated from a synthetic sphere.

**Object recognition approach based on analytic geometry**

Analytically three dimensional objects are a set of two dimensional curves superimposed upon each other. A sphere for example, is superimposed of circles of varying radii. Based upon the 2-D characteristics of standard curves like circles, parabolas, ellipses, and hyperbolas, a unique scheme has been formulated to distinguish standard 3-D objects like spheres, cylinders, cones and ellipsoids.

Each object when intercepted with planes in the horizontal and vertical direction yields a set of curves which is sufficient enough to recognize each of the objects, and at the same time differentiate each from the other.

Consider the equation of a quadric surface,

\[ F(x,y,z) = ax^2 + by^2 + cz^2 + 2fyz + 2gzx + 2hxy + 2px + 2qy + 2rz + d = 0 \]
If this surface is intercepted with a plane parallel to the yz-axis (which means x is a constant), we get an equation of the type

\[ F(x, y, z) = By^2 + Cz^2 + Fyz + Qy + Rz + D = 0 \]

which is an equation of a conic. Based upon the discriminant test [4], which says,

If \[ Ax^2 + Cy^2 + Bxy + Ex + Fy + D = 0 \]
is a equation of a conic, then, based upon the sign of the discriminant, \( B^2 - 4AC \), the curves are of three types.

\[ B^2 - 4AC = 0, \]
implies the curve is a parabola.

\[ B^2 - 4AC < 0, \]
implies the curve is an ellipse.

And finally,

\[ B^2 - 4AC > 0, \]
implies the curve is a hyperbola.

3. Practical Implementation and Experimental Results

Two sets of range data namely, the ones generated using system A and system B is to be experimented with and the following objectives were to be achieved. Each set i.e., A and B are composed of range images of spheres and cylinders respectively.

1. Study the effect of median filtering of different mask sizes on each of the sets.
2. Come up with a method which would evaluate the performance of two different laser range mappers.

Making use of the image processing unit in the Image processing and Computer Vision lab at ODU, range images of objects like sphere and cylinder were segmented
in order to separate the object from the background.

The resulting image which is referred to as the raw image is then median filtered with mask sizes, (a) 3 x 3, (b) 5 x 5, and (c) 7 x 7.

Consider figure 1 which is the actual range image of a sphere (belonging to set A) with its background. Figure 2 is the image after segmentation. The effect of median filtering on figure 2 can be observed in figure 3 (3 x 3 mask), figure 4 (5 x 5 mask) and figure 5 (7 x 7 mask).

The curvature sign map which was discussed in the earlier section, is then used to study the effect of median filtering on the original image shown in figure 2. Determining the first and second derivative with respect to x- and y-axis and comparison of each of these maps will determine whether or not the median filtering has altered the original range image to any extent. Figures 6(a), 6(b), 6(c), and 6(d) are the first and second derivative with respect to x- and y-axis respectively for figure 2. Similarly figures 7(a), 7(b), 7(c), 7(d) and figures 8(a), 8(b), 8(c), 8(d) and figures 9(a), 9(b), 9(c), 9(d) are the first and second derivatives for the figures 3, 4, 5 respectively.

In all of these figures, the sign "+" is assigned to a particular pixel position if the magnitude of the derivative (first or second) of that pixel is greater than the magnitude of the derivative (first or second) of the pixel to its right. Similarly the sign "-" is assigned to a particular pixel position if the magnitude of the derivative (first or second) is lesser than the magnitude of the derivative (first or second) of the pixel to its right. In the case when the magnitudes of the derivatives (first and second) of either pixels is the same, the sign " " (blank) is assigned.

Sign maps which were mentioned before are also generated to check the integrity of the image data before and after the filtering process. Depending upon the
Figure 1. Original range image of the sphere with its background.
Figure 2. Segmented range image of the sphere without its background.
Figure 3. 3 x 3 filtered range image of the sphere.
Figure 4. 5 x 5 filtered range image of the sphere.
Figure 5. 7 x 7 filtered range image of the sphere.
Figure 6(a). First derivative w.r.t x-axis of the original sphere.
Figure 6(b). First derivative w.r.t y-axis of the original sphere.
Figure 6(c). Second derivative w.r.t x-axis of the original cylinder.
Figure 6(d). Second derivative w.r.t y-axis of the original sphere.
Figure 7(a). First derivative w.r.t x-axis of the sphere filtered with a mask size of 3 X 3.
Figure 7(b). First derivative w.r.t y-axis of the sphere filtered with a mask size of 3 X 3.
Figure 7(c). Second derivative w.r.t x-axis of the sphere filtered with a mask size of 3 X 3.
Figure 7(d). Second derivative w.r.t y-axis of the sphere filtered with a mask size of 3 X 3.
Figure 8(a). First derivative w.r.t x-axis of the sphere filtered with a mask size of 5 X 5.
Figure 8(b). First derivative w.r.t y-axis of the sphere filtered with a mask size of 5 x 5.
Figure 8(c). Second derivative w.r.t x-axis of the sphere filtered with a mask size of 5 X 5.
Figure 8(d). Second derivative w.r.t. y-axis of the sphere filtered with a mask size of 5 X 5.
Figure 9(a). First derivative w.r.t x-axis of a sphere filtered with a mask size of 7 x 7.
Figure 9(b). First derivative w.r.t y-axis of a sphere filtered with a mask size of 7 x 7.
Figure 9(c). Second derivative w.r.t x-axis of the sphere filtered with a mask size of 7 x 7.
Figure 9(d). Second derivative w.r.t y-axis of the sphere filtered with a mask size of 7 x 7.
magnitude of the depth value of a pixel and its adjacent neighbor, a '+' or a '-' sign is assigned to the pixel location in the sign map. Figure 10 is the sign map generated for the original raw image data of the sphere. Similarly figures 11, 12, and 13 are the sign maps for the 3 x 3, the 5 x 5, and the 7 x 7 filtered images of the sphere. A careful observation of all these sign maps does suggest that only a small variation has been brought about due to the filtering processes.

Since the main objective of the median filtering is to remove the salt and pepper noise in the range images and thus present a noise free range image for the evaluation of the object coefficients [1], it is seen from figures 3, 4, and 5 that a fine job has been done by all of these filters. However, looking at the curvatures sign maps it is observed that, as the mask size of the filter increases, the curvature maps start looking more and more different than the original. The 3 x 3 filtered image being the most closest to the original raw image can be utilized for further processing and for describing the surface features.

Once the data files are obtained for each of the images which have been filtered, the depth information of each of these files is converted into rectangular coordinate system [1]. These cartesian coordinate information is then utilized for determining the coefficients which describe each of the objects.

Listed in table 1 are the coefficients obtained for the original range image, the 3 x 3 filtered image, the 5 x 5 filtered image and finally the 7 x 7 filtered image of a sphere. At a glance none of these coefficient sets for certain describe a real sphere. The following procedure is adopted to determine which particular set of coefficients best describes the original image data of the object.
Figure 10. Sign map generated for the original raw image of the sphere taking into consideration the magnitude of the depth value at a particular pixel and its neighboring pixel.
Figure 11. Sign map generated for the 3 x 3 filtered image of the sphere taking into consideration the magnitude of the depth value at a particular pixel and its neighboring pixel.
Figure 12. Sign map generated for the 5 x 5 filtered image of the sphere taking into consideration the magnitude of the depth value at a particular pixel and its neighboring pixel.
Figure 13. Sign map generated for the 7 x 7 filtered image of the sphere taking into consideration the magnitude of the depth value at a particular pixel and its neighboring pixel.
TABLE 1

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Raw Image</th>
<th>3 x 3 filtered image</th>
<th>5 x 5 filtered image</th>
<th>7 x 7 filtered image</th>
</tr>
</thead>
<tbody>
<tr>
<td>A, Coeff. of $x^2$</td>
<td>0.3026</td>
<td>0.2211</td>
<td>-0.4860</td>
<td>0.4242</td>
</tr>
<tr>
<td>B, Coeff. of $y^2$</td>
<td>0.2734</td>
<td>0.2802</td>
<td>-0.3291</td>
<td>0.2178</td>
</tr>
<tr>
<td>C, Coeff. of $z^2$</td>
<td>0.6545</td>
<td>0.7747</td>
<td>-0.3338</td>
<td>0.5845</td>
</tr>
<tr>
<td>E, Coeff. of yz</td>
<td>0.5310</td>
<td>-0.5038</td>
<td>0.4834</td>
<td>-0.3417</td>
</tr>
<tr>
<td>F, Coeff. of xz</td>
<td>0.6357</td>
<td>-0.4860</td>
<td>0.7194</td>
<td>-0.7452</td>
</tr>
<tr>
<td>G, Coeff. of xy</td>
<td>0.3524</td>
<td>0.2339</td>
<td>-0.5801</td>
<td>0.4353</td>
</tr>
<tr>
<td>P, Coeff. of x</td>
<td>0.30365</td>
<td>0.19995</td>
<td>-0.3159</td>
<td>0.3127</td>
</tr>
<tr>
<td>Q, Coeff. of y</td>
<td>0.4199</td>
<td>0.4401</td>
<td>-0.3524</td>
<td>0.1996</td>
</tr>
<tr>
<td>R, Coeff. of z</td>
<td>-0.8172</td>
<td>-1.0163</td>
<td>0.3191</td>
<td>-0.5858</td>
</tr>
<tr>
<td>D, Constant</td>
<td>0.2847</td>
<td>0.3717</td>
<td>-0.0973</td>
<td>0.1516</td>
</tr>
</tbody>
</table>
A small surface patch of the object is chosen. In the quadratic form

\[ F(x,y,z) = ax^2 + by^2 + cz^2 + 2fyz + 2gzx + 2hxy + 2px + 2qy + 2rz + d = 0 \]

the coefficients \( a, b, c, d, f, g, h, p, q, \) and \( r \) are inserted and for each \( (x,y,z) \) of the object patch the error is evaluated for each set of coefficients. A plot is thus generated in which every point of the surface patch is replaced with the numerals 1, 3, 5, and 7 signifying that the minimum error was obtained for that particular set of coefficient. Numeral 1 refers to the situation when the original set of coefficients fits best, and similarly numerals 3, 5, and 7 are used depending whether the 3 x 3 or the 5 x 5 or the 7 x 7 set of coefficients give the least error. Figure 14 is one such plot obtained using the coefficients listed in table 1 of the sphere.

The next objective to achieve is that of evaluating the performance of two different laser range mappers. As mentioned before in section 2, the sets A and B consist of two different sets of range images abstracted from two different laser range mappers. For evaluating the performance, the range information of two different spheres obtained from either of these mappers is utilized. Let’s call the range image of the sphere using system A as \( \text{spherel} \). Similarly let’s call the range image of the sphere obtained using system B as \( \text{sphere2} \). A surface patch of \( \text{spherel} \) consisting of 8086 points was selected for experimentation purposes. Similarly the surface patch of \( \text{sphere2} \) had 726 points. Using the approach discussed in section 2 whereby the mean square error is evaluated by trying to a fit a set of data to a real sphere, the mean square errors for \( \text{spherel} \) and \( \text{sphere2} \) is obtained.

Mean square errors are obtained for the raw image, and the 3 x 3 image for \( \text{spherel} \) and \( \text{sphere2} \). The mean square error for the \( \text{spherel} \) belonging to set A was found to be 0.010191 units and 0.009921 units (raw image and 3 x 3 filtered image.
Figure 14. Best fit plot for the sphere belonging to set A. Numerals "1, 3, and 5" denote the original sphere, 3 x 3 filtered image, and 5 x 5 filtered image respectively.
respectively). The mean square errors for sphere2 belonging to set B was found to be .019095 units and 0.018686 units (raw and 3 x 3 filtered images respectively).

The curvature maps for sphere and cylinder belonging to the sets A and B are shown in appendix A. Appendix B lists out the ten coefficients obtained for all the different images of sets A and B. Files with extension *.cod serve as the input for the program evaluating the coefficients, and the files with extension *.coe consists of the output data, which are the needed necessary coefficients. Appendix C consists of a detailed listing of all the programs utilized.

4. CONCLUSIONS

In this research, range images of objects obtained using laser range mappers are utilized to recognize three dimensional regular objects. Due to inherent problems in the laser range mappers, the depth information obtained by itself cannot be utilized to make a accurate description of the object. The approach involving the evaluations of the ten coefficients which best describe an object is utilized on filtered images of the original objects. Inspite of using noise free images, it is seen that the coefficients obtained for each object does not infer the shape of any of the objects.

A new approach which involves 2-D analytical geometry has been discussed briefly which appears very promising for the recognition of 3-D objects. The coefficients obtained earlier do come in handy while using a discriminant test for describing each of the objects with 2-D curves. In the future research the above new theory formulated will be utilized for making a accurate description of each of the regular 3-D objects.

Calculations evaluating the performance of the two different laser range mappers
quite distinctly showed that laser range mapper A performs better than laser range mapper B.
LIST OF REFERENCES


APPENDIX A

Curvature sign maps of the following range images is included in this appendix.

1. Original cylinder image belonging to set A.
2. 3 x 3 filtered image of the cylinder belonging to set A.
3. 5 x 5 filtered image of the cylinder belonging to set A.
4. 7 x 7 filtered image of the cylinder belonging to set A.
5. Original sphere image belonging to set B.
6. 3 x 3 filtered image of the sphere belonging to set B.
7. 5 x 5 filtered image of the sphere belonging to set B.
8. Original cylinder image belonging to set B.
9. 3 x 3 filtered image of the cylinder belonging to set B.
10. 5 x 5 filtered image of the cylinder belonging to set B.

For each of the above images the curvature sign maps consists of the first and second derivatives with respect to the x- and y-axis. Sets A and B signify to the images mapped by two different laser range mappers.
Images belonging to set A
First derivative w.r.t x-axis of the original cylinder.
First derivative w.r.t y-axis of the original cylinder.
Second derivative w.r.t x-axis of the original cylinder.
Second derivative w.r.t y-axis of the original cylinder.
First derivative w.r.t x-axis of the cylinder filtered with a mask size of 3 X 3.
First derivative w.r.t y-axis of the cylinder filtered with a mask size of 3 X 3.
Second derivative w.r.t x-axis of the cylinder filtered with a mask size of 3 X 3.
Second derivative w.r.t y-axis of the cylinder filtered with a mask size of 3 X 3.
First derivative w.r.t x-axis of the cylinder filtered with a mask size of 5 X 5.
First derivative w.r.t y-axis of the cylinder filtered with a mask size of 5 X 5.
Second derivative w.r.t x-axis of the cylinder filtered with a mask size 5 X 5.
Second derivative w.r.t y-axis of the cylinder filtered with a mask size of 5 X 5.
First derivative w.r.t x-axis of the cylinder filtered with a mask size of 7 X 7.
First derivative w.r.t y-axis of the cylinder filtered with a mask size of 7 X 7.
Second derivative w.r.t x-axis of the cylinder filtered with a mask size of 7 X 7.
Second derivative w.r.t y-axis of the cylinder filtered with a mask size of 7 X 7.
Images belonging to set B
First derivative w.r.t x-axis of the original sphere.
First derivative w.r.t y-axis of the original sphere.
Second derivative w.r.t x-axis of the original sphere.
Second derivative w.r.t y-axis of the original sphere.
First derivative w.r.t the x-axis of the sphere filtered with a mask size of 3 x 3.
First derivative w.r.t y-axis of the sphere filtered with a mask size of 3 X 3.
Second derivative w.r.t. x axis of the sphere filtered with a mask size of 3 X 3.
Second derivative w.r.t y-axis of the sphere filtered with a mask size of 3 X 3.
First derivative w.r.t x-axis of the sphere filtered with a mask size of 5 X 5.
First derivative w.r.t y-axis of the sphere filtered with a mask size of 5 X 5.
Second derivative w.r.t x-axis of the sphere filtered with a mask size of 5 X 5.
Second derivative w.r.t y-axis of the sphere filtered with a mask size of 5 X 5.
First derivative w.r.t y-axis of the original cylinder.
First derivative w.r.t x-axis of the original cylinder.
Second derivative w.r.t x-axis of the original cylinder.
Second derivative w.r.t y-axis of the original cylinder.
First derivative w.r.t x-axis of the cylinder filtered with a mask size of 3 X 3.
First derivative w.r.t y-axis of the cylinder filtered with a mask size of 3 X 3.
Second derivative w.r.t x-axis of the cylinder filtered with a mask size of 3 X 3.
Second derivative w.r.t y-axis of the cylinder filtered with a mask size of 3 X 3.
First derivative w.r.t y-axis of the cylinder filtered with a mask size of 5 X 5.
First derivative w.r.t x-axis of the cylinder filtered with a mask size of 5 X 5.
Second derivative w.r.t x-axis of the cylinder filtered with a mask size of 5 X 5.
Second derivative w.r.t y-axis of the cylinder filtered with a mask size of 5 X 5.
APPENDIX B

This appendix consists of the ten coefficients generated for the original and processed range images of a sphere and cylinder mapped using two different laser range mappers. Files with extension *.cod consists of range data converted into cartesian coordinates, and the files with extension *.coe consists of the coefficients generated for each of the images.
The input file was "sprawl.cod"
The output file is "sprawl.coe"
The coeff of x-squared is 0.3026157
The coeff of y-squared is 0.2734349
The coeff of z-squared is 0.6545654
The coeff of yz is -0.5310194
The coeff of zx is -0.6357662
The coeff of xy is 0.3524517
The coeff of x is 0.3036514
The coeff of y is 0.4199182
The coeff of z is -0.8172019
The constant d is 0.2847408

Coefficients of the original sphere image belonging to group A.
The input file was "spraw31.cod"
The output file is "spraw31.coe"
The coeff of x-squared is  0.2211579
The coeff of y-squared is  0.2802473
The coeff of z-squared is  0.7747064
The coeff of  yz     is -0.5038247
The coeff of  zx     is -0.4860164
The coeff of  xy     is  0.2339016
The coeff of  x      is  0.1995363
The coeff of  y      is  0.4401489
The coeff of  z      is -1.016356
The constant  d      is  0.3717703

Coefficients of the 3 x 3 filtered image of the sphere belonging to group A.
The input file was "spraw51.COD"
The output file is "spraw51.COE"
The coeff of $x^2$ is -0.4860452
The coeff of $y^2$ is -0.3291118
The coeff of $z^2$ is -0.3338964
The coeff of $yz$ is 0.4834592
The coeff of $zx$ is 0.7194569
The coeff of $xy$ is -0.5801437
The coeff of $x$ is -0.3159497
The coeff of $y$ is -0.3524498
The coeff of $z$ is 0.3191445
The constant $d$ is -9.7348504E-02

Coefficients of the 5 x 5 filtered image of the sphere belonging to group A.
The input file was "sprawme1.coe"
The output file is "sprawme1.coe"
The coeff of x-squared is 0.4242373
The coeff of y-squared is 0.2178874
The coeff of z-squared is 0.5845248
The coeff of yz is -0.3417171
The coeff of zx is -0.7452961
The coeff of xy is 0.4353395
The coeff of x is 0.3127908
The coeff of y is 0.1996729
The coeff of z is -0.5858592
The constant d is 0.1516084

Coefficients of the 7 x 7 filtered image of the sphere belonging to group A.
The input file was "cyraw1.cod"
The output file is "cyraw1.coe"
The coeff of x-squared is 0.1555596
The coeff of y-squared is 0.2353804
The coeff of z-squared is 0.8288453
The coeff of yz is -0.6818960
The coeff of zx is 3.7034817E-02
The coeff of xy is 2.1725880E-02
The coeff of x is -0.2105054
The coeff of y is 0.5823037
The coeff of z is -1.317142
The constant d is 0.5681907

Coefficients of the original cylinder belonging to group A.
The input file was "cyraw31.cod"
The output file is "cyraw31.coe"
The coeff of x-squared is 0.2676638
The coeff of y-squared is 0.1930158
The coeff of z-squared is 0.7483451
The coeff of \( yz \) is -0.5485628
The coeff of \( zx \) is 0.5481051
The coeff of \( xy \) is -0.2466192
The coeff of \( x \) is -0.7515414
The coeff of \( y \) is 0.5662742
The coeff of \( z \) is -1.360964
The constant \( d \) is 0.6880789

Coefficients of the 3 x 3 filtered image of the cylinder belonging to group A.
The input file was "cyraw51.cod"
The output file is "cyraw51.coe"
The coeff of x-squared is 5.4338872E-02
The coeff of y-squared is 9.9206299E-02
The coeff of z-squared is 0.2060992
The coeff of yz is -0.1109364
The coeff of zx is 1.265334
The coeff of xy is -0.5254330
The coeff of x is -1.185869
The coeff of y is 0.3039300
The coeff of z is -0.7311586
The constant d is 0.5089003

Coefficients of the 5 x 5 filtered image of the cylinder belonging to group A.
The input file was "cyrawme1.cod   
The output file is "cyrawme1.coe   
The coeff of x-squared is   0.1532317
The coeff of y-squared is   -9.9520542E-02
The coeff of z-squared is   -0.4889523
The coeff of yz is   0.4767834
The coeff of zx is   1.008621
The coeff of xy is   -0.4587431
The coeff of x is   -1.006533
The coeff of y is   -0.2328676
The coeff of z is   0.4734453
The constant d is   -1.3768099E-02

Coefficients of the 7 x 7 filtered image of the cylinder belonging to group A.
The input file was "R3SPHERE.COD"
The output file is "R3SPHERE.COE"
The coeff of x-squared is 0.1027336
The coeff of y-squared is 3.8939383E-02
The coeff of z-squared is 0.5696317
The coeff of yz is 0.6472183
The coeff of zx is -0.9516000
The coeff of xy is -4.9645115E-02
The coeff of x is 0.8889613
The coeff of y is -0.6169493
The coeff of z is -1.387174
The constant d is 0.7926015

Coefficients of the original sphere belonging to group B.
The input file was "R3SPHER3.COD  "
The output file is "R3SPHER3.COE  "
The coeff of x-squared is  -4.9067583E-02
The coeff of y-squared is   0.4412566
The coeff of z-squared is   0.6636547
The coeff of  yz        is  -1.2313786E-02
The coeff of  zx        is   -0.5175490
The coeff of  xy        is  -0.6759338
The coeff of  x         is   0.6078625
The coeff of  y         is  -0.3856263
The coeff of  z         is  -1.276605
The constant  d        is   0.6796699

Coefficients of the 3 x 3 filtered image of the sphere belonging to group B.
The input file was "R3SPHER5.COD"
The output file is "R3SPHER5.COE"
The coeff of x-squared is -5.7173960E-02
The coeff of y-squared is -0.1170360
The coeff of z-squared is 0.5475225
The coeff of yz is 0.5604561
The coeff of zx is 1.006907
The coeff of xy is 0.1962991
The coeff of x is -1.006644
The coeff of y is -0.3863406
The coeff of z is -0.9040802
The constant d is 0.3384019

Coefficients of the 5 x 5 filtered image of the sphere belonging to group B.
The input file was "R6CYLIN.COD"
The output file is "R6CYLIN.COE"
The coeff of x-squared is 0.9754460
The coeff of y-squared is 2.5132844E-02
The coeff of z-squared is 3.5924029E-02
The coeff of yz is -6.8559073E-02
The coeff of zx is 3.1578626E-02
The coeff of xy is 0.2957501
The coeff of x is 0.2924450
The coeff of y is 0.1052131
The coeff of z is -1.9418295E-02
The constant d is 1.5252778E-02

Coefficients of the original cylinder belonging to group B.
The input file was "R6CYLIN3.COD"
The output file is "R6CYLIN3.COE"
The coeff of x-squared is -4.7388867E-02
The coeff of y-squared is -0.3104874
The coeff of z-squared is -0.3682815
The coeff of yz is 1.192302
The coeff of zx is 0.1264399
The coeff of xy is -0.3063811
The coeff of x is -2.7492255E-02
The coeff of y is -0.9607195
The coeff of z is 0.3220469
The constant d is 4.0601194E-03

Coefficients of the 3 x 3 cylinder image belonging to group B.
The input file was "R6CYLIN5.COD"
The output file is "R6CYLIN5.COE"
The coeff of x-squared is 1.7619731E-02
The coeff of y-squared is 0.7016529
The coeff of z-squared is -0.2045088
The coeff of yz is -0.3910733
The coeff of zx is -0.7922655
The coeff of xy is -0.3879120
The coeff of x is 0.8651381
The coeff of y is -0.1430389
The coeff of z is 0.2737453
The constant d is 1.2079749E-02

Coefficients of the 5 x 5 filtered image of the cylinder belonging to group B.
APPENDIX C

This appendix consists of the listings of the following programs.

1. Program which performs the 3 x 3 and 5 x 5 median filtering.

2. Program that evaluates the first and second derivative w.r.t to x- and y-axis of the data files and then transforms it into a sign map.

3. Program that generates the sign map for each of the range images based upon the magnitude of range value of neighboring pixels. Sign maps for the cylinder of set A and the sphere and cylinder of set B are included at the end of the listing.

4. Program that generates a numeral map based upon the evaluation of the least square errors from the generated coefficients.

5. Program that generates the ten coefficients which describes each of the range images.
C**** PROGRAM MEDIAN FILTERING

C**** THIS PROGRAM PERFORMS THE MEDIAN FILTERING ON THE
C**** ORIGINAL RANGE IMAGE FILES. BY CHANGING THE
C**** PARAMETER "M". A 3x3 OR A 5x5 MASK SIZE CAN BE UTILIZED
C**** FOR FILTERING.

PARAMETER (N=512)
INTEGER*2 A(N,N),MED(N,N)
CHARACTER*12 INFILE,OUTFILE

C MAIN PROGRAM
C
WRITE(*,123)
123 FORMAT(5X,'INPUT FILE NAME : INFILE')
READ(*,*)INFILE
WRITE(*,223)
223 FORMAT(5X,'OUTPUT FILENAME : OUTFILE')
READ(*,*)OUTFILE

OPEN (UNIT = 1,FILE = INFILE,RECL = 2048,STATUS = 'OLD')
READ (1,9)((A(I,J),J = 1,N),I = 1,N)
9 FORMAT(512I4)
M=3
CLOSE(1,DISPOSE = 'SAVE')
CALL MEDFLT(A,MED,N,M)
OPEN (UNIT = 2,FILE = OUTFILE,RECL = 2048,STATUS = 'NEW')
WRITE (2,11)((MED(I,J),J = 1,N),I = 1,N)
11 FORMAT(512I4)
CLOSE(2,DISPOSE = 'SAVE')
STOP
END

SUBROUTINE MEDIAN FILTER
SUBROUTINE MEDFLT(A,MED,N,M)
INTEGER*2 A(N,N),MED(N,N),SORT(50)
LOGICAL NEXCHAN

MM=M ** 2
X=(M+1)/2
Y=X-1
M1=(MM+1)/2
DO 7 I=X,(N-Y)
DO 9 J=X,(N-Y)
K1=0
DO 11 K=(I-Y),(I+Y)
DO 13 L = (J-Y),(J+Y)
    K1 = K1 + 1
    SORT(K1) = A(K,L)
13 CONTINUE
11 CONTINUE
DO 15 II = 1,(MM-1)
DO 17 K1 = 1,(MM-II)
    IF (SORT(K1).GT.SORT(K1+1)) THEN
        TEMP = SORT(K1)
        SORT(K1) = SORT(K1+1)
        SORT(K1+1) = TEMP
    END IF
17 CONTINUE
15 CONTINUE
MED(I,J) = SORT(M1)
9 CONTINUE
7 CONTINUE
DO 19 I = 1,Y
    DO 21 J = 1,N
        MED(I,J) = A(I,J)
        MED(N+1-I,J) = A(N+1-I,J)
        MED(J,N+1-I) = A(J,N+1-I)
        MED(J,I) = A(J,I)
21 CONTINUE
19 CONTINUE
RETURN
END
C***** PROGRAM DERIVATIVES

C***** THIS PROGRAM DETERMINES THE DERIVATIVES
C***** ALONG THE X-AXIS AND THE Y-AXIS. A GROUP OF FILES CAN BE
C***** COMPARED TO SEE WHETHER A PARTICULAR LOCATION HAS THE SAME
C***** CURVATURE OR NOT.

    INTEGER*2 I1,J1,T1,P1,K,L,I,J
    REAL    DX1,DX2,DX3,DY1,DY2,DY3
    REAL    DX11,DX22,DX33,DY11,DY22,DY33
    REAL    D(70,350),E(70,350),A(1000,3),AA(60,50)
    REAL    D1(70,350),E1(70,350)
    CHARACTER*12 INFILE1,INFILE2,INFILE3,POINT
    CHARACTER*2 GRAPH1(70,100),GRAPH2(70,100),GRAPH3(70,100)
    CHARACTER*2 GRAPH4(70,100)

WRITE(*,20)
FORMAT(5X,'INPUT FILE NAME : INFILE1')
READ(*,*)INFILE1
OPEN(UNIT = 1, FILE = INFILE1, STATUS = 'UNKNOWN')

WRITE(*,25)
FORMAT(5X,'INPUT TOTAL # OF PTS : N1')
READ(*,*)N1
DO 100 I = 1,N1
    READ(1,*)(A(I,J),J = 1,3)
CONTINUE

DO 811 K = 1,51
    DO 815 L = 1,19
        AA(K,L) = A(L + (19*(K-1)),3)
    CONTINUE

103 FORMAT(51214)
TO FIND THE DERIVATIVE ALONG X-AXIS

C1111 WRITE(*,908)
C908 FORMAT('INPUT THE STARTING RECORD NUMBER: STREC')
C READ(*,*)STREC
C WRITE(*,9008)
C9008 FORMAT('INPUT THE ENDING RECORD NUMBER: ENDREC')
C READ(*,*)ENDREC

OPEN(UNIT = 2,FILE = 'FILE1.X',STATUS = 'UNKNOWN')
OPEN(UNIT = 3,FILE = 'FILE1.Y',STATUS = 'UNKNOWN')
OPEN(UNIT = 4,FILE = 'FILE1.XX',STATUS = 'UNKNOWN')
OPEN(UNIT = 8,FILE = 'FILE1.YY',STATUS = 'UNKNOWN')

11178 DO 1104 I1 = 1,51
    DO 1204 J1 = 1,19
        D(I1,J1) = 0.5*((AA(I1,J1+1)-AA(I1,J1))+(AA(I1+1,J1+1)-AA(I1+1,J1)))
    CONTINUE

103
\[ D_{1(I,J)} = (A_{A(I,J-1)} - 2 \times A_{A(I,J)} + A_{A(I,J+1)}) \]

\[ E_{1(I,J)} = (A_{A(I+1,J)} - 2 \times A_{A(I,J)} + A_{A(I-1,J)}) \]

\[ E_{1(I,J)} = 0.5 \times (A_{A(I,J+1)} - A_{A(I,J-1)}) + (A_{A(I+1,J)} - A_{A(I-1,J)}) \]

CONTINUE

DO 1104 \( I = 1,51 \)
WRITE(2,*)\( D(I,J), J = 1,19 \)
WRITE(3,*)\( E(I,J), J = 1,19 \)
WRITE(4,*)\( D(I,J), J = 1,19 \)
WRITE(8,*)\( E(I,J), J = 1,19 \)
CONTINUE

CLOSE(2)
CLOSE(3)
CLOSE(4)
CLOSE(8)

OPEN\( UNIT = 2, FILE = 'FILE1.X', STATUS = 'UNKNOWN' \)
OPEN\( UNIT = 3, FILE = 'FILE1.Y', STATUS = 'UNKNOWN' \)
OPEN\( UNIT = 4, FILE = 'FILE1.XX', STATUS = 'UNKNOWN' \)
OPEN\( UNIT = 5, FILE = 'FILE1.YY', STATUS = 'UNKNOWN' \)

DO 324 \( I = 1,51,1 \)
READ(2,*)\( D(I,J), J = 1,19 \)
CONTINUE

CONTINUE

DO 325 \( I = 1,51,1 \)
DO 326 \( J = 1,19 \)
IF \( D(I,J).GT.D(I,J+1) \) THEN
GRAPH1\( (I,J) = ' - ' \)
ENDIF
IF \( D(I,J).LT.D(I,J+1) \) THEN
GRAPH1\( (I,J) = ' + ' \)
ENDIF
IF \( D(I,J).EQ.D(I,J+1) \) THEN
GRAPH1\( (I,J) = ' ' \)
ENDIF
CONTINUE

CONTINUE

DO 328 \( I = 1,51,1 \)
READ(3,*)\( D(I,J), J = 1,19 \)
CONTINUE

CONTINUE

DO 329 \( I = 1,51,1 \)
DO 330 \( J = 1,19 \)
IF \( D(I,J).GT.D(I,J+1) \) THEN
GRAPH2\( (I,J) = ' - ' \)
ENDIF
IF \( D(I,J).LT.D(I,J+1) \) THEN
GRAPH2\( (I,J) = ' + ' \)
ENDIF
IF \( D(I,J).EQ.D(I,J+1) \) THEN
GRAPH2\( (I,J) = ' ' \)
ENDIF
ENDIF
CONTINUE
DO 332 I1 = 1,51,1
READ(4,*)(E(I1,J1),J1 = 1,19)
CONTINUE
DO 332 I1 = 1,51,1
DO 334 J1 = 1,19
  IF (E(I1,J1).GT.E(I1,J1+1)) THEN
    GRAPH3(I1,J1) = '.'
  ENDIF
  IF (E(I1,J1).LT.E(I1,J1+1)) THEN
    GRAPH3(I1,J1) = '+'
  ENDIF
  IF (E(I1,J1).EQ.E(I1,J1+1)) THEN
    GRAPH3(I1,J1) = ','
  ENDIF
CONTINUE
DO 336 I1 = 1,51,1
READ(5,*)(EI(I1,J1),J1 = 1,19)
CONTINUE
DO 336 I1 = 1,51,1
DO 338 J1 = 1,19
  IF (EI(I1,J1).GT.EI(I1,J1+1)) THEN
    GRAPH4(I1,J1) = '.'
  ENDIF
  IF (EI(I1,J1).LT.EI(I1,J1+1)) THEN
    GRAPH4(I1,J1) = '+'
  ENDIF
  IF (EI(I1,J1).EQ.EI(I1,J1+1)) THEN
    GRAPH4(I1,J1) = ','
  ENDIF
CONTINUE
CONTINUE
CONTINUE
CONTINUE
DO 21104 I1 = 1,51,1
WRITE(13,1234)(GRAPH1(I1,J1),J1 = 1,19)
WRITE(14,1234)(GRAPH2(I1,J1),J1 = 1,19)
WRITE(15,1234)(GRAPH3(I1,J1),J1 = 1,19)
WRITE(16,1234)(GRAPH4(I1,J1),J1 = 1,19)
CONTINUE
FORMAT(30X,20A1)
WRITE(*,21)
GOTO 64
END
C**** PROGRAM RANGE SIGN MAP

C***** THIS PROGRAM GENERATES A SIGN MAP FOR DATA FILES BY TAKING
C***** INTO CONSIDERATION THE ABSOLUTE DIFFERENCE IN RANGE VALUE
C***** OF NEIGHBORING PIXELS.

INTEGER*2 A(0:511,0:512),D(100,100)
INTEGER*2 I1,J1,I1,P1,ZZ,XX
CHARACTER*12 INFILE1,INFILE2,INFILE3,POINT
CHARACTER*2 GRAPH1(100,100)
WRITE(*,20)
FORMAT(5X,'INPUT FILE NAME : INFILE1')
READ(*,*)INFILE1
OPEN(UNIT=1, FILE=INFILE1, STATUS='UNKNOWN', RECL=2048)
DO 100 I=1,511
READ(1,300)(A(I,J),J = 1,512)
CONTINUE
100 FORMAT(512I4)
ZZ=I
XX=I
DO 43 I = 165,215
XX=I
DO 53 J = 260,278
D(ZZ,XX)=A(I,J)
ZZ=ZZ+I
XX=XX+I
CONTINUE
XX=I
ZZ=ZZ+I
XX=I
CONTINUE
WRITE(*,*)XX,ZZ

C**** TEST FILE USED FOR THIS PROGRAM IS THAT OF THE CYLINDER
C**** BELONGING TO SET A.

OPEN(UNIT=2,FILE='RANGEVAL.DAT',STATUS='UNKNOWN')
OPEN(UNIT=3,FILE='RANGEDIFF.DAT',STATUS='UNKNOWN')
OPEN(UNIT=4,FILE='FILE1.XX',STATUS='UNKNOWN')

DO 325 I=1,ZZ-1
DO 326 J=1,XX-1
IF (D(I,J),GT.D(I,J+1))THEN
GRAPH1(I,J) = '+'
ENDIF
IF (D(I,J),LT.D(I,J+1))THEN
GRAPH1(I,J) = '.'
ENDIF
106
ENDIF
IF (D(I,J),EQ,D(I,J+1)) THEN
GRAPH1(I,J) =  ''
ENDIF
CONTINUE
CONTINUE
DO 21104  I = 1,ZZ-1
WRITE(3,1234)(GRAPH1(I,J), J = 1,XX-1)
WRITE(2,3000)(D(I,J), J = 1,XX-1)
21104 CONTINUE
1234 FORMAT(35X,20A1)
3000 FORMAT(14)
STOP
END
C**** PROGRAM BEST FIT COEFFICIENTS

C**** THIS PROGRAM MAKES A PLOT USING THE COEFFICIENTS GENERATED
C**** FROM THE PROGRAM 'SURFACE.FOR'. AT EACH PIXEL OF A TEST
C**** SURFACE PATCH, THE ERROR IS DETERMINED USING THE GENERATED
C**** COEFFICIENTS OF THE ORIGINAL RANGE DATA, THE 3X3 RANGE IMAGE,
C**** THE 5X5 RANGE IMAGE, AND THE 7X7 RANGE IMAGE. WHICHEVER
C**** GIVES THE MINIMUM ERROR REPLACES THE PIXEL WITH THE NUMERAL
C**** 1, 3, 5, 7 WHEREEVER APPLICABLE.

C**** TEST FILE IN THE PROGRAM ARE THE RANGE IMAGES OF THE
C**** CYLINDER BELONGING TO GROUP A.

REAL A(5000,3),B(5000,3),C(5000,3),D(5000),H(5000,3)
REAL E(5000),F(5000),P(5000)
INTEGER G(5000),PLOT(100,100)

OPEN(UNIT=1,FILE='CYRAW1.PLT',STATUS='UNKNOWN')
OPEN(UNIT=2,FILE='CYRAWME1.PLT',STATUS='UNKNOWN')
OPEN(UNIT=3,FILE='CYRAW51.PLT',STATUS='UNKNOWN')
OPEN(UNIT=4,FILE='CY RAW31.PLT',STATUS='UNKNOWN')
OPEN(UNIT=8,FILE='CYLINDE2.PLT',STATUS='UNKNOWN')

DO 10 I = 1,969
  READ(I,*) (A(I,J), J = 1,3)
  CONTINUE
DO 40 I = 1,969
  DO 50 J = 1,3
    D(I) = (0.15555*A(I,1)*A(I,1)) + (.23538*A(I,2)*A(I,2)) +
           (0.8288*A(I,3)*A(I,3)) - (0.6818*A(I,2)*A(I,3)) +
           (0.03703*A(I,1)*A(I,3)) + (0.021725*A(I,1)*A(I,2)) -
           (0.2105*A(I,1)) + (0.58230*A(I,2)) -
           (1.317142*A(I,3)) + (0.568190)
  CONTINUE
DO 20 I = 1,969
  READ(2,*) (B(I,J), J = 1,3)
  CONTINUE
DO 50 I = 1,969
  DO 50 J = 1,3
    E(I) = (0.15323*B(I,1)*B(I,1)) - (0.09952*B(I,2)*B(I,2)) -
           (0.48895*B(I,3)*B(I,3)) + (0.47678*B(I,2)*B(I,3)) +
           (1.00862*B(I,1)*B(I,3)) - (0.4587431*B(I,1)*B(I,2)) -
           (1.006533*B(I,1)) - (0.23286*B(I,2)) +
           (0.473445*B(I,3)) - (0.013768)
  CONTINUE
DO 30 I = 1,969
  READ(3,*) (C(I,J), J = 1,3)
  CONTINUE
DO 60 I = 1,969
  DO 60 J = 1,3
    F(I) = (0.15042*C(I,1)*C(I,1)) + (0.09952*C(I,2)*C(I,2)) +
           (0.48895*C(I,3)*C(I,3)) - (0.47678*C(I,2)*C(I,3)) -
           (1.00862*C(I,1)*C(I,3)) + (0.4587431*C(I,1)*C(I,2)) +
           (1.006533*C(I,1)) + (0.23286*C(I,2)) -
           (0.473445*C(I,3)) + (0.013768)
  CONTINUE
DO 80 I = 1,969
  READ(4,*) (P(I,J), J = 1,3)
  CONTINUE
DO 90 I = 1,969
  DO 90 J = 1,3
    G(I) = (0.15555*P(I,1)*P(I,1)) + (0.23538*P(I,2)*P(I,2)) +
           (0.8288*P(I,3)*P(I,3)) - (0.6818*P(I,2)*P(I,3)) +
           (0.03703*P(I,1)*P(I,3)) + (0.021725*P(I,1)*P(I,2)) -
           (0.2105*P(I,1)) + (0.58230*P(I,2)) -
           (1.317142*P(I,3)) + (0.568190)
  CONTINUE
DO 100 I = 1,969
  READ(5,*) (H(I,J), J = 1,3)
  CONTINUE
DO 110 I = 1,969
  DO 110 J = 1,3
    H(I) = (0.15323*H(I,1)*H(I,1)) - (0.09952*H(I,2)*H(I,2)) -
           (0.48895*H(I,3)*H(I,3)) + (0.47678*H(I,2)*H(I,3)) +
           (1.00862*H(I,1)*H(I,3)) - (0.4587431*H(I,1)*H(I,2)) -
           (1.006533*H(I,1)) - (0.23286*H(I,2)) +
           (0.473445*H(I,3)) - (0.013768)
  CONTINUE
DO 120 I = 1,969
  READ(6,*) (PLOT(I,J), J = 1,100)
  CONTINUE
DO 130 I = 1,969
  READ(7,*) (PLOT(I,J), J = 1,100)
  CONTINUE
DO 140 I = 1,969
  READ(8,*) (PLOT(I,J), J = 1,100)
  CONTINUE

C

108
\[ F(I) = (0.054338 \times C(I,1) \times C(I,1)) + (0.099206 \times C(I,2) \times C(I,2)) + (0.206092 \times C(I,3) \times C(I,3)) - (0.110936 \times C(I,2) \times C(I,3)) + (1.265334 \times C(I,1) \times C(I,3)) - (0.525433 \times C(I,1) \times C(I,2)) - (1.18586 \times C(I,1)) + (0.303930 \times C(I,2)) - (0.7311586 \times C(I,3)) + (0.5089003) \]

\[ \text{CONTINUE} \]

\[ \text{DO 301 } I = 1,969 \]
\[ \text{READ}(4,*) (H(I,J),J = 1,3) \]
\[ \text{CONTINUE} \]

\[ \text{DO 602 } I = 1,969 \]
\[ \text{DO 50 } J = 1,3 \]
\[ P(I) = (0.26766 \times H(I,1) \times H(I,1)) + (0.193015 \times H(I,2) \times H(I,2)) + (0.7483451 \times H(I,3) \times H(I,3)) - (0.548105 \times H(I,2) \times H(I,3)) + (0.548105 \times H(I,1) \times H(I,3)) - (0.246619 \times H(I,1) \times H(I,2)) - (0.751541 \times H(I,1)) + (0.5662742 \times H(I,2)) - (1.360964 \times H(I,3)) + (0.6880789) \]
\[ \text{CONTINUE} \]

\[ \text{DO 90 } I = 1,969 \]
\[ \text{IF}((D(I) \lt E(I)) \text{.AND.} (D(I) \lt F(I)) \text{.AND.} (D(I) \lt P(I)) \text{.THEN}} \]
\[ G(I) = 1 \]
\[ \text{ENDIF} \]
\[ \text{ENDIF} \]

\[ \text{IF}((E(I) \lt D(I)) \text{.AND.} (E(I) \lt F(I)) \text{.AND.} (E(I) \lt P(I)) \text{.THEN}} \]
\[ G(I) = 7 \]
\[ \text{ENDIF} \]
\[ \text{ENDIF} \]

\[ \text{IF}((F(I) \lt E(I)) \text{.AND.} (F(I) \lt D(I)) \text{.AND.} (F(I) \lt P(I)) \text{.THEN}} \]
\[ G(I) = 5 \]
\[ \text{ENDIF} \]
\[ \text{ENDIF} \]

\[ \text{ELSE} \]
\[ \text{ELSE} \]
\[ \text{ELSEIF}((P(I) \lt E(I)) \text{.AND.} (P(I) \lt D(I)) \text{.AND.} (P(I) \lt F(I)) \text{.THEN}} \]
\[ G(I) = 3 \]
\[ \text{ENDIF} \]
\[ \text{ELSE} \]
\[ \text{ELSEIF}((D(I) \lt E(I)) \text{.AND.} (D(I) \lt F(I)) \text{.AND.} (D(I) \lt E(I)) \text{.THEN}} \]
\[ G(I) = 4 \]
\[ \text{ENDIF} \]
\[ \text{ENDIF} \]
\[ \text{ELSEIF}((D(I) \lt F(I)) \text{.AND.} (E(I) \lt F(I)) \text{.THEN}} \]
\[ \text{IF}(D(I) \lt E(I)) \text{.THEN} \]
\[ G(I) = 9 \]
\[ \text{ENDIF} \]
\[ \text{ELSEIF}((D(I) \lt F(I)) \text{.AND.} (E(I) \lt F(I)) \text{.THEN}} \]
\[ \text{IF}(D(I) \lt E(I)) \text{.THEN} \]
\[ G(I) = 4 \]
\[ \text{ENDIF} \]
\[ \text{ELSEIF}((D(I) \lt F(I)) \text{.AND.} (E(I) \lt F(I)) \text{.THEN}} \]
\[ \text{IF}(D(I) \lt E(I)) \text{.THEN} \]\n\[ G(I) = 4 \]
\[ \text{ENDIF} \]

109
G(I)=6
ENDIF
ENDIF
IF((F(I).LT.D(I)).AND.(E(I).LT.D(I)))THEN
IF(F(I).EQ.E(I))THEN
G(I)=8
ENDIF
ENDIF
CONTINUE
DO 1000 I = 1,51
DO 2000 J = 1,19
PLOT(I,J) = G(J + 19*(I-1))
2000 CONTINUE
1000 CONTINUE
C
DO 3000 I = 1,51
DO 4000 J = 1,42
WRITE(8,5000)(PLOT(I,J),J = 1,19)
3000 CONTINUE
5000 format(20x,19i1)
stop
end
Best fit plot obtained for the cylinder belonging to set A. Numerals "1, 3, 5, 7" denote the original image, the 3 x 3 image, the 5 x 5 image, and the 7 x 7 image respectively.
Best fit plot for the sphere belonging to set B.
Numerals "3, 5" denote the filtered 3 x 3 and 5 x 5 images of the original sphere.
Best fit plot for the cylinder belonging to set B. Numerals "1, 3, 5" denote the original cylinder image, the 3 x 3 image, and the 5 x 5 image.
C***** PROGRAM SURFACE

C **************************************************************
C THIS PROGRAM APPROXIMATES THE COEFFICIENTS OF A SURFACE
C GENERATED BY GIVEN DATA POINTS. THE INPUT FILE IS 'DATA.DAT'
C CONSISTING OF COORDINATES OF POINTS ON SOME SURFACE.
C **************************************************************

INTEGER I,J,K,IP
REAL X(9000),Y(9000),Z(9000),X_2(9000)
REAL YZ(9000),ZX(9000),XY(9000),P_PTR(9000,10,10),SC(10,10)
REAL A(4,4),B(6,4),B_TR(4,6),C(6,6),H(6,6),H_INV(6,6)
REAL RIS(4,8),A_INV(4,4),BA_INV(6,4),BA_INVBT(6,6),M(6,6)
REAL H_INVM(6,6),M_PR(6,6),AI(6,6),BI(6,6),CI(6,6)
REAL EIGVAL(6,6),EIGVEC(6,6),E_VEC(6),A_INVBT(4,6)
REAL ALPHA(4),BETA(6),A_VECT(10)
CHARACTER*18 INFILE,OUTFILE

WRITE(*,3)
3 FORMAT(SX,'INPUT TOTAL POINTS NOT EXCEEDING 7750: IP=')
READ(*,*) IP
ROOT = 1/(SQRT(2.))
DO 24 I = 1,6
DO 26 J = 1,6
H(I,J) = 0
26 CONTINUE
24 CONTINUE
H(1,1) = 1
H(2,2) = 1
H(3,3) = 1
H(4,4) = ROOT
H(5,5) = ROOT
H(6,6) = ROOT
C
ROOT1 = SQRT(2.)
DO 20 I = 1,6
DO 22 J = 1,6
H_INV(I,J) = 0
22 CONTINUE
20 CONTINUE
H_INV(1,1) = 1
H_INV(2,2) = 1
H_INV(3,3) = 1
\[ H_\text{INV}(4,4) = \text{ROOT1} \]

\[ H_\text{INV}(5,5) = \text{ROOT1} \]

\[ H_\text{INV}(6,6) = \text{ROOT1} \]

C****** DATA IS READ HERE ********************

DO 30 I = 1,IP
  READ(l,*) (X(I),Y(I),Z(I))
30    CONTINUE

C****** THE VECTOR P FOR SCATTER MATRIX IS FORMED HERE ******

DO 32 I = 1,IP
  X_2(I) = X(I)**2
  Y_2(I) = Y(I)**2
  Z_2(I) = Z(I)**2
  YZ(I) = Y(I)*Z(I)
  ZX(I) = Z(I)*X(I)
  XY(I) = X(I)*Y(I)
32    CONTINUE

DO 34 I = 1,IP
  P(I,1) = X_2(I)
  P(I,2) = Y_2(I)
  P(I,3) = Z_2(I)
  P(I,4) = YZ(I)
  P(I,5) = ZX(I)
  P(I,6) = XY(I)
  P(I,8) = V(I)
  P(I,9) = Z(I)
  P(I,10) = 1
34    CONTINUE

DO 36 J = 1,10
  DO 38 K = 1,10
    P_PTR(I,J,K) = P(I,J)*P(I,K)
 38    CONTINUE
36    CONTINUE

DO 42 J = 1,10
  DO 44 K = 1,10
    SC(J,K) = 0
44    CONTINUE
42    CONTINUE

C***** THE SCATTER MATRIX IS FORMED HERE  ******************

DO 46 J = 1,10
  DO 48 K = 1,10
    SC(J,K) = SC(J,K) + P_PTR(I,J,K)
48    CONTINUE
46    CONTINUE

115
CONTINUE
CONTINUE

THE SCATTER MATRIX SC IS DECOMPOSED INTO A,B,B_TR,C

DO 52 I = 1,6
  DO 54 J = 1,6
    C(I,J) = SC(I,J)
  CONTINUE
52 CONTINUE

DO 56 I = 1,6
  DO 58 J = 1,4
    B(I,J) = SC(I,J+6)
  CONTINUE
56 CONTINUE

DO 60 I = 1,4
  DO 62 J = 1,6
    A_L(J) = SC(I,J+6)
  CONTINUE
60 CONTINUE

DO 64 I = 1,4
  DO 66 J = 1,4
    A_INV(I,J) = RIS(I,J)
  CONTINUE
64 CONTINUE

DO 68 I = 1,6
  DO 70 J = 1,4
    RIS(I,J) = A(I,J)
  CONTINUE
68 CONTINUE

CALL INVERS(RIS,4,4,A_INV)

DO 72 I = 1,4
  DO 74 J = 1,4
    A_INV(I,J) = RIS(I,J)
  CONTINUE
72 CONTINUE

DO 76 I = 1,6
  DO 78 J = 1,4
    BA_INV(I,J) = 0
  CONTINUE
76 CONTINUE

DO 80 I = 1,6
  DO 82 J = 1,4
    DO 84 K = 1,4
      BA_INV(I,J) = BA_INV(I,J) + B(I,K)*A_INV(K,J)
    CONTINUE
80 CONTINUE

DO 86 I = 1,6
  DO 88 J = 1,4
    BA_INV(I,J) = 0
  CONTINUE
86 CONTINUE

CONTINUE
PREVIOUS PAGE
CONTINUE
DO 90 I = 1,6
   DO 92 J = 1,6
      DO 94 K = 1,4
         BA_INVBT(I,J) = BA_INVBT(I,J) + BA_INV(I,K) * B_TR(K,J)
      CONTINUE
   CONTINUE
90 CONTINUE
CONTINUE
DO 96 I = 1,6
   DO 98 J = 1,6
      M(I,J) = C(I,J) - BA_INVBT(I,J)
   CONTINUE
96 CONTINUE
98 CONTINUE
C ******** NOW TO COMPUTE M' **************
C
DO 100 I = 1,6
   DO 102 J = 1,6
      H_INVM(I,J) = 0
   CONTINUE
100 CONTINUE
DO 104 I = 1,6
   DO 106 J = 1,6
      DO 108 K = 1,6
         H_INVM(I,J) = H_INVM(I,J) + H_INV(I,K) * M(K,J)
      CONTINUE
106 CONTINUE
104 CONTINUE
DO 110 I = 1,6
   DO 112 J = 1,6
      M_PR(I,J) = 0
   CONTINUE
110 CONTINUE
DO 114 I = 1,6
   DO 116 J = 1,6
      DO 118 K = 1,6
         M_PR(I,J) = M_PR(I,J) + H_INVM(I,K) * H_INV(K,J)
      CONTINUE
116 CONTINUE
114 CONTINUE
118 CONTINUE
C ******** NOW TO FIND THE EIGEN VALUES OF M' **************
C
ND = 6
CALL EIG(ND, M_PR, EIGVAL, EIGVEC)
C
C ******** TO FIND THE SMALLEST EIGEN VALUE AND ITS CORRESPONDING **
C ******** EIGEN VECTOR ****************************************
C
S_EIG = EIGVAL(1,1)
KOUNT = 1
DO 120 I=2,6
   IF (S_EIG.GT.EIGVAL(I,I)) THEN
      S_EIG=EIGVAL(I,I)
      KOUNT=I
   ENDIF
120 CONTINUE
DO 122 I=1,6
   EI_VEC(I)=EIGVEC(I,KOUNT)
122 CONTINUE
DO 124 I=1,6
   BETA(I)=0
   DO 126 J=1,6
      BETA(I)=BETA(I)+H_INV(I,J)*EI_VEC(J)
126 CONTINUE
124 CONTINUE
DO 128 I=1,4
   DO 130 J=1,6
      A_INVBT(I,J)=0
   DO 132 K=1,4
      A_INVBT(I,J)=A_INVBT(I,J)+A_INV(I,K)*BTR(K,J)
132 CONTINUE
130 CONTINUE
128 CONTINUE
136 CONTINUE
DO 138 I=1,6
   A_VEC(I)=BETA(I)
138 CONTINUE
134 CONTINUE
DO 140 I=1,4
   DO 142 J=1,6
      WRITE(2,*) (' THE COEFF OF ZX IS ',A_VECT(5))
      WRITE(2,*) (' THE COEFF OF XY IS ',A_VECT(6))
      WRITE(2,*) (' THE COEFF OF X IS ',A_VECT(7))
      WRITE(2,*) (' THE COEFF OF Y IS ',A_VECT(8))
      WRITE(2,*) (' THE COEFF OF Z IS ',A_VECT(9))
      WRITE(2,*) (' THE CONSTANT D IS ',A_VECT(10))
142 CONTINUE
130 CONTINUE
C142 CONTINUE
CLOSE(UNIT=2,DISPOSE='SAVE')
CLOSE(UNIT=1,DISPOSE='SAVE')
SUBROUTINE INVERS(RIS,N,NX,MX)
DIMENSION RIS(NX,MX)
N1=N-1
N2=2*N
DO 2 I=1,N
   DO 1 J=1,N
      J1=J+N
1   RIS(I,J1)=0.
      J1=I+N
2   RIS(I,J1)=1.
   DO 10 K=1,N1
      C=RIS(K,K)
      IF (ABS(C)-0.000001) 3,3,5
      K1=K+1
      DO 6 J=K1,N2
9   RIS(K,J)=RIS(K,J)/C
      DO 10 I=K1,N
      C=RIS(I,K)
      DO 10 J=K1,N2
8   RIS(I,J)=RIS(I,J)-C*RIS(K,J)
10 CONTINUE
5   NPI=N1+1
   IF (ABS(RIS(N,N))-0.000001) 3,3,19
19   DO 20 J=NP1,N2
20   RIS(N,J)=RIS(N,J)/RIS(N,N)
   DO 200 L=1,N1
      K=N-L
      K1=K+1
   DO 200 I=NP1,N2
      DO 200 J=K1,N
200   RIS(K,I)=RIS(K,I)-RIS(K,J)*RIS(J,I)
   DO 250 I=1,N
      J1=I+N
250   RIS(I,J1)=RIS(I,J1)
RETURN
3   TYPE*, 'SINGULARITY IN ROW FOUND'
RETURN
END

SUBROUTINE EIG(ND,AL,BI,CI)
DIMENSION AL(ND,ND),BI(ND,ND),CI(ND,ND)
INTEGER N1,M1,N2,M2
N1=ND
M1=ND
N2=ND
M2=ND
ANORM = 0.0
SN = FLOAT(N2)
DO 100 I = 1, N2
    DO 101 J = 1, N2
        IF (I-J) 72, 71, 72
71    BI(I,J) = 1.0
        GOTO 101
72    BI(I,J) = 0.0
        ANORM = ANORM + AI(I,J) * AI(I,J)
101    CONTINUE
100    CONTINUE
ANORM = SQRT(ANORM)
FNORM = ANORM * (1.0E-09/SN)
THR = ANORM
23    THR = THR/SN
3    IND = 0
    DO 102 I = 2, N2
        II = I-1
        DO 103 J = 1, II
            IF (ABS(AI(J,I)) - THR) 103, 4, 4
4            IND = 1
            AL = -AI(J,I)
            AM = (AI(J,J) - AI(I,I))/2.0
            AO = AL/SQRT((AL*AL) + (AM*AM))
            IF (AM) 5, 6, 6
5            AO = -AO
6            SINV = AO/SQRT(2.0*(1.0 + SQRT(1.0 - AO*AO)))
            SINX2 = SINV*SINV
            COSX = SQRT(1.0 - SINX2)
            COSX2 = COSX*COSX
            DO 104 K = 1, N2
                IF (K-J) 7, 8, 10
7                AT = AI(K,J)
                AI(K,J) = AT*COSX - AI(K,I)*SINV
                AI(K,J) = AT*SINX + AI(K,I)*COSX
8                BT = BI(K,J)
                BI(K,J) = BT*COSX - BI(K,I)*SINV
                BI(K,J) = BT*SINX + BI(K,I)*COSX
104            CONTINUE
    XT = 2.0*AI(J,I)*SINV*COSX
    AT = AI(J,J)
    BT = AI(I,I)
    AI(J,J) = AT*COSX2 + BT*SINV2 - XT
    AI(I,I) = AT*SINV2 + BT*COSX2 + XT
    AI(J,I) = (AT-BT)*SINV*COSX + AI(J,I)*(COSX2-SINV2)
    AI(I,J) = AI(I,J)
    DO 105 K = 1, N2
        AI(J,K) = AI(K,J)
        AI(I,K) = AI(K,I)
105    CONTINUE
103    CONTINUE
102 CONTINUE
   IF (IND) 20,20,3
20    IF (THR-FNORM) 25,25,23
25       DO 110 I=2,N2
26       J=1
29       IF ((ABS(AI(J-1,J-1)))-(ABS(AI(J,J)))) 30,110,110
30          AT = AI(J-1,J-1)
101         AI(J-1,J-1) = AI(J,J)
102         AI(J,J) = AT
103         DO 111 K=1,N2
104            AT = BI(K,J-1)
105            BI(K,J-1) = BI(K,J)
106            BI(K,J) = AT
111     CONTINUE
112     J=J-1
113     IF (J-1) 110,110,29
110    CONTINUE
114    DO 112 I=1,N2
115       DO 114 J=1,N2
116          CI(I,J) = BI(I,J)
117       BI(I,J) = AI(I,J)
118    CONTINUE
119    CONTINUE
120    RETURN
121 END
C
C
The effect of filtering processes on range images is studied and the performance of two different laser range mappers is evaluated. Median filtering is utilized to remove noise from the range images. First and second order derivatives are then utilized to locate the similarities and dissimilarities between the processed and the original images. Range depth information is converted into spatial coordinates, and a set of coefficients which describe three-dimensional objects is generated. Range images of spheres and cylinders are used for experimental purposes. An algorithm is developed to compare the performance of two different laser range mappers based upon the range depth information of surfaces generated by each of the mappers. Furthermore, an approach based on two-dimensional analytic geometry, which serves as a basis for the recognition of regular three-dimensional geometric objects is also proposed.