Range Image Segmentation with Applications to Robot Bin-Picking Using Vacuum Gripper

EZZET AL-HUJAZI AND ARUN SOOD, SENIOR MEMBER, IEEE

Abstract —**Widespread interest in factory automation and robotics has spurred research in the field of machine vision. Most previous research work has used intensity images. In the last few years range data analysis has become very popular. Range data acquisition is slower than that of intensity images. However, depth information allows us to handle a wide variety of applications more efficiently. A new segmentation algorithm that can be used for robot applications is presented. The input images are dense range data of industrial parts. The image is segmented into a number of surfaces. The segmentation algorithm uses the residual analysis to detect edges. Then a region growing technique is used to obtain the final segmented image. Most of the research in 3-D vision has centered on recognizing objects. The use of the segmentation output for determining the best holdsite position and orientation is studied. As compared to techniques based on intensity images the use of range images simplifies the holdsite determination. This information can then be used to instruct the robot to grip the object and move it to the required position. The performance of the algorithm on a number of range images is presented.**

I. Introduction

IN AUTONOMOUS SYSTEM OPERATION the use of imaging sensors is necessary for supporting planning, control and safety functions. The use of imaging sensors enables autonomous systems to achieve higher level of autonomy and to handle a variety of tasks. The problem of image analysis for autonomous operation is complex and requires fast, reliable, inexpensive and widely applicable methods. A wide variety of imaging sensors has been utilized in research and manufacturing environments. Recently, range sensing has become very popular. Active range sensors can be classified into two categories: structured light, and time of flight. Detailed description of these active range sensors are given in [5], [26]. Active sensors are attractive for autonomous system applications because they provide depth information without the computation overhead associated with passive techniques such as stereo. On the other hand, active range sensors are not suitable for specular surfaces. In our research we have performed our experiments on dense range data obtained using a laser range finder, though we expect that these techniques will apply to depth map data obtained using other sensors.

In this paper we focus on a surface based segmentation of dense range images of industrial objects that can be used for a wide variety of applications. The approach combines edge and region growing techniques to segment images. The algorithm begins by detecting edges based on the difference between the image and a smoothed version. In our analysis we have utilized a Gaussian smoothing filter. Following [24], we will refer to the analysis of the difference as the residual analysis. However, as compared to that paper, we use a slightly different function of residual. We show that the absolute value of the residual (AR) in one-dimension (1-D) is maximum, at the edge point, for step and roof edges. The resulting edge points are then linked to form a closed connected components. In the final step, an iterative region growing technique is used to obtain the final segmentation.

Three-dimensional imaging techniques have applications in a variety of robot tasks, for example, recognition, planning and manipulation. A recent survey paper [26] outlines the structure of robot sensing systems and the techniques for measuring and preprocessing of three-dimensional (3-D) data. 3-D robot applications can be classified into two major categories: man-made objects and natural objects. Scenes including man-made objects are easier to analyze, and most of the 3-D vision techniques that have been developed are applicable to such objects. Man-made objects can be represented by a set of primitives that is well defined mathematically (e.g., surfaces, lines, points). Man-made objects include indoor objects like those found in industrial, office and household environments and outdoor objects (e.g., roads, cars and airplanes). Representation of natural objects (e.g.,

Manuscript received June 12, 1989; revised March 6, 1990. This work has been supported in part by research grants from Belvoir Research Development and Engineering Center under contract number DAAK 70-89-K-0027 and Emhart Planning Research Corporation and in part by NSF equipment grant CDA-8906889. A preliminary version of this paper was presented at SPIE Applications of Artificial Intelligence VII Conference.

E. Al-Hujazi is with the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI 48202.

A. K. Sood is with the Department of Computer Science, George Mason University, Fairfax, VA 22030.

IEEE Log Number 9037595.

0018-9472/90/1100-1313$01.00 ©1990 IEEE

trees, rivers and hills) is a much more complex problem than representation of man-made objects.

Most 3-D vision research has been directed toward object recognition and path planning. The problem of 3-D object recognition is very difficult because it requires a detailed representation to allow for the discrimination of similar objects. There are a number of issues that must be considered in designing a 3-D object recognition system: 1) what and how many objects to consider, 2) what viewing conditions to choose, 3) what primitives to utilize, 4) how to construct the object model from primitives, 5) what matching algorithm to use. For a mobile robot system [18], an important task of 3-D vision is to provide sufficient information to the path planner. For example, the vision system can be used for detecting obstacles and recognizing landmarks, and accordingly update the initial representation used for the path planner.

Due to the complexity of the recognition and the path planning problems, it is anticipated that multiple sensors and multiple features will be required to perform these tasks more efficiently. In [18], the authors have discussed strategies to combine range information with color. A typical application for this combination involves the use of color for road following and range information for obstacle avoidance. In [21], an approach that combines photometric stereo and binocular stereo as vision input tools is presented. Photometric stereo is used to determine the location and orientation of the object. The binocular stereo system is used to produce a coarse elevation map for detecting a collision-free configuration for the gripper and for measuring the height at the selected pickup point. It is emphasized that in these applications depth information play a critical role.

A number of applications require that the robot have the capability to manipulate an object without having to recognize it. The computational complexity of this problem is considerably less than that of object recognition. To manipulate an object, gripping point(s) must be found. The shape and size of the holdsite will depend on the robot hand employed. For example parallel-jaw hand needs two positions on opposing surfaces for holdsites. On the other hand, vacuum gripper requires a single surface holdsite.

The range image segmentation technique presented in this paper can be utilized to develop a surface representation for a wide variety of applications. However, in order to be specific we utilize an example from the robot manipulation category to demonstrate the application of our methodology. We will show how the results of our segmentation algorithm can be directly used in determining good holdsites for gripping a part in a bin by a robot. The use of our segmentation algorithm for this application is direct and straightforward. Based on the output of our segmentation algorithm and the objects we are considering, we have chosen a contour-adapting vacuum gripper [28] for the robot hand. Such a hand requires that the holdsite meet some minimal requirements. For a surface area to qualify as a holdsite it must be greater than a specific minimal size and the surface depth variation in the holdsite area should be small. The holdsite must be well inside the surface away from any periphery of the surface. Use of range images and the proposed segmentation algorithm provides a convenient method of achieving these constraints.

The remainder of this paper is divided into seven sections. In Section II several methods that have been proposed for robot bin picking are discussed. Section III gives a brief review of existing range image segmentation techniques. The mathematical derivation of the AR edge detection approach and the segmentation algorithm are given in Sections IV and V, respectively. Section VI presents experimental results of applying the segmentation algorithm on a number of range images. A procedure for detecting potential holdsites from the segmented images is given in Section VII. Finally, some conclusions and extensions of the presented method are given in Section VIII.

II. Robot Bin Picking Methods

A number of manufacturing tasks are being automated using industrial robots. In the simplest case a robot, without vision sensing, can pick up an object from a bin by moving around the pile until an object is acquired. Different kind of hands have been proposed for such a task [28]. These approaches become increasingly inadequate as the probability of meeting potential object holdsite decreases. This probability can be improved by using proximity sensors. In most applications, however, these approaches are inadequate because of the time delay in locating an object for gripping.

A vision sensor, integrated with an industrial robot, greatly increases the variety of applications that can be implemented and can improve the speed of operation. However, in a bin where overlapping objects are possible, vision systems have limited capabilities to recognize the object and estimate object pose [29]. On the other hand, isolated objects on a flat background usually have a small number of stable states and hence are relatively easy to analyze. The problem of identifying an object or determining object pose in a pile can be performed by segmenting the topmost object from the pile. The topmost object is then moved, placed against suitable background and constrained in degrees of freedom (maybe dropped on a planar surface). The object can then be analyzed to determine its identity and its pose.

A typical robot vision system is shown in Fig. 1. The data is first acquired using, for example, intensity or range sensors. In the second step the data is processed to extract useful information to control the robot. If the information obtained by the second stage is not sufficient for the robot to perform the task, iterative sensing will be undertaken (feedback loops in Fig. 1).

The problem of determining good grip points requires that location, height, size and orientation of a visible surface be determined. Thereafter in a visible surface a

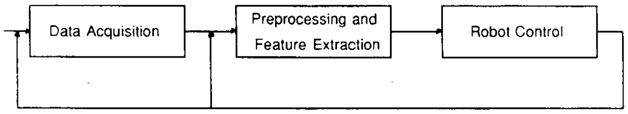


Fig. 1. Typical robot vision system.

potential holdsite is determined. A number of algorithms based on intensity images are available in the literature. Dessimoz *et al.* [12] presented a matched filter algorithm that detects potential holdsites for objects based on a parallel-jaw robot hand. Bin picking by detecting brightness changes has been treated in [8]. Ikeuchi *et al.* [21] presented an approach that uses photometric stereo to segment the image. One of the segmented regions is then used as the target region. Range information, not available from photometric stereo, is obtained by the Prism stereo system. A set of potential grip configurations is first extracted from models. Then based on the observed data a grip configuration is selected. The range data is then used to find a collision-free configuration. The authors considered the parallel-jaw gripper. As shown in Section VIII the use of range data simplifies the problem of locating potential holdsites.

III. Range Data Segmentation

Segmentation of an image into regions of similar properties is essential for higher level computer vision processing. These regions can be used to generate a representation of images by defining attributes and relations between regions. Such a representation can in turn be used in recognition, planning or manipulation tasks. Object recognition methods involve comparison with a stored model [11], [23]. Detailed representation is needed to allow for the discrimination among similar objects. Coarse representation [14] is usually sufficient for other tasks such as obstacle avoidance or object manipulation.

Approaches to 3-D object representation can be classified into three types based on the dimensionality of the primitives used: approaches that use 3-D volumetric primitives, 2-D surface primitives, or 1-D line primitives. Volumetric representation [25] divides the 3-D object into a set of 3-D volume elements. A large number of elements are usually required and number of views are necessary to generate the representation. Boundary representation [16] uses edge and point information. Due to the difficulty in manipulating and matching general spatial curve functions, most of the research in this area makes the assumption that edges will be linear or circular. Surface representation [20] involves segmenting the object into a number of surfaces and then describe each surface. The relations between surfaces are then defined to generate the representation. Problems arise when certain views of an object provides little information to make a unique identification. A number of researchers have based their work on simple surface primitives (e.g., planar, spherical, etc.). This is because of the difficulties associated with manipulation and parameter stability of general quadratic functions in the presence of noise. Surface representation is the most widely used representation for range data and can be readily obtained by additional processing of the output of our segmentation algorithm. Surface representation is the most suitable one for holdsite determination since it provides us with the accessible surfaces in the object and their descriptions.

Several solutions for range image segmentation have been proposed in the literature. These techniques fall into two classes: 1) region growing or clustering and 2) edge detection techniques. Region growing techniques attempt to group pixels into connected regions based on similarity measures. Edge detection techniques locate pixels that lie on the boundary between two regions.

Region growing methods that are based on function approximation of the surface are commonly used in range image analysis. Some researchers require that region pixels be well approximated by planar or quadric surfaces [7], [19]. Other approaches are geared toward segmenting cylinders in range data [1]. Hebert and Ponce [17] segmented planes, cylinders and cones from range data. Oshima and Shirai [27] used planes, spheres, cylinders and cones. Many of these methods obtain an initial segmentation of small primitive regions and then iteratively merge these regions until all merges (allowed by a smoothness or approximation error constraint) have taken place.

To obtain initial seed regions, concepts from differential geometry have been utilized in extracting surface characteristics from range data. Segmentation based on differential geometric quantities, such as lines of curvature or surface curvature, requires region pixels to have similar geometric properties. Differential geometry of surfaces analyzes the local differences of surface points. Most theorems of differential geometry address global topological similarities. Global shape similarity theorems exist for convex objects. Difficulties arise when local descriptors are used to identify the shape and global similarities of nonconvex objects in range data. Besl and Jain [6] proposed the use of H (mean curvature) and K (Gaussian curvature) curvature information to extract seed regions. These regions are then grown and merged using a variable-order surface fit to obtain homogeneous segmented surfaces. The authors show good segmentation results of their algorithm on a number of range and intensity images, but the control structure of the algorithm is very complex. In [2] an algorithm for segmenting range images based on a first-order Markov Random Field was presented. The algorithm computes H and K curvature parameters at each pixel and a deterministic

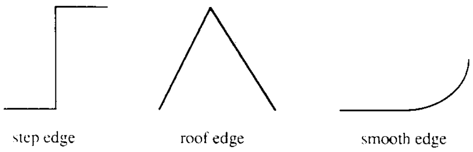


Fig. 2. Edge types in range images.

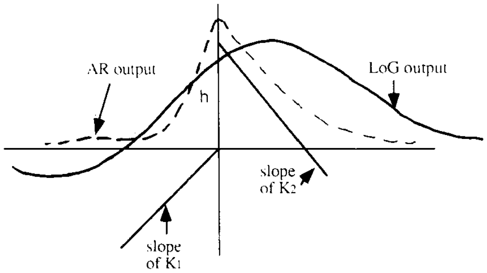


Fig. 3. Output of the Laplacian of Gaussian and AR for asymmetric step edge ($K_1 = 1$, $K_2 = 4$, $h = 10$, and $\sigma = 3$).

region growing algorithm is then used to segment the image.

Clustering technique for range image segmentation have been proposed in [20]. The algorithm minimizes a square error criterion. The problem with clustering techniques is that the number of regions, which is required by the clustering algorithm, is not known in advance. Specifying a large number of regions will result in over segmenting the image. To overcome this problem merging of adjacent regions must be performed.

Edge detection techniques have played an important role in range image analysis. There are three basic primitive types of edges in range images, step, roof (crease) and smooth edges. Most of the research work has concentrated on the first two types of edges. Step edges correspond to discontinuities in depth values. Roof edges correspond to surface creases where the surface normals are discontinuous. Smooth edges are characterized by continuity of surface normals but discontinuity of curvature. Detection of roof and smooth edges is very difficult because they do not correspond to large depth variation and tend to hide in noise. Fig. 2 shows 1-D examples of the three edge types.

A variety of methods are available for detecting edges in range images. Step edges are usually detected using techniques similar to those of intensity images [10]. Most of these approaches are based on first or second gradient computation which is sensitive to noise. Prefiltering the data may help in some cases, but this introduces uncertainty in the location of the edge. To illustrate the problems in applying conventional edge detection techniques to range images we consider the application of a Laplacian of Gaussian (LoG) operator with a standard deviation of σ to an asymmetric step edge (a combination of step and roof edges) that occur frequently in range images (see for example Fig. 10 and Fig. 12). Fig. 3 shows the delocalization introduced by the LoG operator when applied to such an edge. In the figure k_1 and k_2 are the slopes of the lines forming the edge and h is the step size. Since the step and roof edges can be derived as special cases of this asymmetric edge we find it convenient to use this as the edge model in the next section.

Detection of roof and smooth edges is a more difficult problem. Inokuchi *et al.* [22] present an edge detection algorithm for range data using a ring operator. The ring operator extracts a 1-D periodic function of depth values that surround a given pixel. The function is transformed to the frequency domain using an FFT algorithm. Jump and roof edges are then detected by examining the frequency components of the ring surrounding the pixel. A multiscale approach to extract information for planar contours was proposed in [13]. The curve is repeatedly smoothed with Gaussian masks and features are tracked in different scales. Brady *et al.* [9] extract and analyze a dense set of 3-D lines of curvature to detect and describe surface intersections. This method is computationally complex, and takes a long time to even describe simple objects.

Recently, residual analysis has been suggested for detecting edges. This approach uses the difference between the input image and the smoothed image to detect edge points. This is the basis of the algorithm presented in Section IV. The residual analysis was informally introduced in [15] in the context of visual surface reconstruction. It has also been used by Lee and Pavlidis [24] to detect step edges in range data using a regularized smoothing filter. Fig. 3 also shows the result of AR computation. As can be seen from the figure AR did not introduce any localization error for this edge.

IV. Our Approach to Range Data Segmentation

In the context of range images, edge detection based image segmentation techniques have difficulty in detecting roof edges. On the other hand, region growing techniques require good initial seed regions for their success. We propose an approach that combines edge detection and region growing techniques. A preliminary version of the algorithm was presented in [3]. The edge points are detected first using AR. The edge points obtained are then linked and connected components are formed. Iterative region growing is then used to obtain the final segmented image. In this section we present the mathematical analysis and the performance of edge detection using AR. A detailed description of the algorithm is given in Section V.

A. Mathematical Derivation

The mathematical derivation of AR for step and roof edges is based on the 1-D edge model shown in Fig. 4. In the figure, h is the step height and is zero for a roof edge,

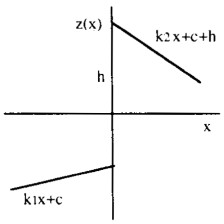


Fig. 4. Edge model.

c is a constant and k_1, k_2 are the slopes of the lines forming the edge. We will show here that AR has a maxima at the edge point for all k_1, k_2, h and c. The mathematical analysis presented here is based on the use of Gaussian smoothing filter. Let $z(x)$ be the range image depth value at point x, and let $z_S(x)$ be the smoothed depth value. The edge model $z(x)$ is defined as:

$$z(x)=\begin{cases} k_1x+c & x<0 \\ k_2x+c+h & x\geqslant 0 \end{cases}. \tag{1}$$

For ease of presentation we assume that $k_2 \leqslant k_1$. Similar analysis can be done for $k_2 > k_1$. The result of smoothing $z(x)$ with a Gaussian filter is given by

$$z_S(x)=c+h\int_{-\infty}^{x}G(t)\,dt+xk_1\int_{x}^{\infty}G(t)\,dt + xk_2\int_{-\infty}^{x}G(t)\,dt+\frac{k_2-k_1}{\sqrt{2\pi}}\sigma\exp\frac{-x^2}{2\sigma^2} \tag{2}$$

where $G(x)$ is the Gaussian distribution given by

$$G(x)=\frac{1}{\sqrt{2\pi\sigma^2}}\exp\frac{-x^2}{2\sigma^2}$$

and σ is the standard deviation of the Gaussian filter.

We define AR as

$$\mathrm{AR}(x)=|z_S(x)-z(x)|. \tag{3}$$

We are now in a position to state and prove a theorem that is the basis of the algorithm developed in this paper.

Theorem: The value of AR(x) defined by (3) is maximum at $x=0$ for the edge model $z(x)$ defined by (1).

Proof: For ease of presentation we deal with $x \geqslant 0$ and $x<0$ cases separately. For $x \geqslant 0$ substituting from (1) and (2) into (3) yields:

$$\mathrm{AR}(x)=\left|-h\int_{x}^{\infty}G(t)\,dt+\frac{(k_2-k_1)\sigma}{\sqrt{2\pi}}\cdot\left[\exp\frac{-x^2}{2\sigma^2}-\frac{x\sqrt{2\pi}}{\sigma}\int_{x}^{\infty}G(t)\,dt\right]\right|.$$

Using the lemma in the Appendix, we note that the term in the square bracket is positive and since $k_2 < k_1$ both terms on the right hand side are negative. AR can be re-written as:

$$\mathrm{AR}(x)=h\int_{x}^{\infty}G(t)\,dt+\frac{|k_2-k_1|\sigma}{\sqrt{2\pi}}\cdot\left[\exp\frac{-x^2}{2\sigma^2}-\frac{x\sqrt{2\pi}}{\sigma}\int_{x}^{\infty}G(t)\,dt\right]. \tag{4}$$

Once again using the lemma of the Appendix and noting that for $x \geqslant 0$, $\int_x^\infty G(t)\,dt \leqslant 1/2$, we obtain an upper bound on AR:

$$\mathrm{AR}(x)\leqslant\frac{h}{2}+\frac{|k_2-k_1|\sigma}{\sqrt{2\pi}}.$$

It is emphasized that the equality holds at $x=0$ i.e.,

$$\mathrm{AR}(0)=\frac{h}{2}+\frac{|k_2-k_1|\sigma}{\sqrt{2\pi}}. \tag{5}$$

For $x<0$ AR is given by

$$\mathrm{AR}(x)=\left|h\int_{-\infty}^{x}G(t)\,dt+\frac{(k_2-k_1)\sigma}{\sqrt{2\pi}}\cdot\left[\exp\frac{-x^2}{2\sigma^2}+\frac{x\sqrt{2\pi}}{\sigma}\int_{-\infty}^{x}G(t)\,dt\right]\right|. \tag{6}$$

Using the triangle inequality (6) becomes:

$$\mathrm{AR}(x)\leqslant\left|h\int_{-\infty}^{x}G(t)\,dt\right| +\left|\frac{(k_2-k_1)\sigma}{\sqrt{2\pi}}\left[\exp\frac{-x^2}{2\sigma^2}+\frac{x\sqrt{2\pi}}{\sigma}\int_{-\infty}^{x}G(t)\,dt\right]\right|.$$

The first term has an upper bound of $h/2$ and from the lemma in the Appendix the second term has an upper bound of $(|k_2-k_1|\sigma/\sqrt{2\pi})$, and the value of AR for $x<0$ is

$$\mathrm{AR}(x)<\frac{h}{2}+\frac{|k_2-k_1|\sigma}{\sqrt{2\pi}}=\mathrm{AR}(0).$$

Hence, AR(x) $\leqslant$ AR(0) for all x. This completes the proof of the theorem.

We now consider two special cases. First, consider a step edge with $k_1 = k_2 = 0$, AR is given by

$$\mathrm{AR}(x)=\begin{cases} h\int_{-\infty}^{x}G(t)\,dt & x<0 \\ h\int_{x}^{\infty}G(t)\,dt & x\geqslant 0 \end{cases} \tag{7}$$

which has a maxima at $x=0$ with

$$\mathrm{AR}(0)=h/2. \tag{8}$$

For a roof edge with $h = 0$ AR at given by:

$$\mathrm{AR}(x) = \begin{cases} \dfrac{|k_2 - k_1|\sigma}{\sqrt{2\pi}} \left[\exp \dfrac{-x^2}{2\sigma^2} + \dfrac{x\sqrt{2\pi}}{\sigma} \displaystyle\int_{-\infty}^{x} G(t)\,dt \right] & x < 0 \\ \dfrac{|k_2 - k_1|\sigma}{\sqrt{2\pi}} \left[\exp \dfrac{-x^2}{2\sigma^2} - \dfrac{x\sqrt{2\pi}}{\sigma} \displaystyle\int_{x}^{\infty} G(t)\,dt \right] & x \geqslant 0 \end{cases}$$

which has a maxima at $x = 0$. The value of $\mathrm{AR}(x)$ at zero is given by

$$\mathrm{AR}(0) = \frac{|k_2 - k_1|\sigma}{\sqrt{2\pi}}. \tag{9}$$

Equation (8) shows that the value of AR is independent of the standard deviation of the smoothing filter for a step edge, while (9) shows that the value of AR for a roof edge can be increased by increasing the standard deviation of the Gaussian smoothing filter. The curvature difference must be high to obtain a reliable estimate of roof edge points. This suggests that a small filter is sufficient to detect the step edge while a large filter will improve the detection of roof edges.

B. Performance Analysis

The performance of an edge detector is usually evaluated using the following criterion.

1) Output signal-to-noise ratio (SNR): There should be a low probability of failing to mark real edge points, and low probability of falsely marking nonedge points. Since both these probabilities are monotonically decreasing functions of SNR, this criterion corresponds to maximizing SNR.
2) Good localization: The points marked as edge points should be as close as possible to the location of the true edge.

Similar measures are used in [10].

1) SNR: The SNR of the AR approach is evaluated as follows: Let the impulse response of the operator be denoted by $f(x)$. The operator used to evaluate AR is defined by

$$f(x) = G(x) - \delta(x) \tag{10}$$

where $\delta(x)$ is the Dirac delta function.

The response of the operator to $z(x)$, at the edge point $(x = 0)$, is given by

$$S = \int_{-W}^{W} z(-t) f(t)\,dt \tag{11}$$

where the operator has a finite impulse response bounded by $[-W, W]$. The root-mean-squared response to the noise $n(x)$ only is

$$N = n_o \left[\int_{-W}^{W} f^2(t)\,dt \right]^{1/2} \tag{12}$$

TABLE I

σ	sum
0.5	0.137
1.0	0.713
2.0	0.924
3.0	0.97
4.0	0.98
5.0	0.98

where n_o^2 is the mean square noise amplitude per unit length. SNR is then given by

$$\mathrm{SNR} = \mathrm{S/N} = \frac{\left| \int_{-W}^{W} z(-t) f(t)\,dt \right|}{n_o \sqrt{\int_{-W}^{W} f^2(t)\,dt}}. \tag{13}$$

Using $f(x)$ defined in (10), N is given by

$$N = n_o \left[\int_{-W}^{W} (G(t) - \delta(t))^2\,dt \right]^{1/2}. \tag{14}$$

Equation 14 can be approximated using discrete analysis. The response to noise is then given by

$$N^2 = n_o^2\,(\mathrm{sum})$$

where

$$\mathrm{sum} = \sum_{\substack{i=-W \\ i \neq 0}}^{W} G(i)^2 + \sum_{i=0} (G(i) - 1)^2.$$

The sum part of the previous equation was evaluated for different values of σ as shown in Table 1. The sum was evaluated using a window of size $4\sigma + 1$. The table shows that $N \leqslant n_o$ for the values of σ shown in the table. Once again we find that for high values of σ, N can be approximated by n_o. We use this approximation in the subsequent analysis.

The value of S for a step edge is equal to $h/2$ (8). Hence, the SNR is given by:

$$\mathrm{SNR}_{\text{step}} \approx \frac{h/2}{n_o}. \tag{15}$$

This shows that the SNR for the step edge is independent of the smoothing filter size. Thus a small filter is sufficient to detect step edges.

For a roof edge S is given in (9), and the SNR is

$$\mathrm{SNR}_{\text{roof}} \approx \frac{|k_2 - k_1|\sigma}{n_o\sqrt{2\pi}}. \tag{16}$$

Thus SNR for a roof edge can be increased by increasing the filter size. The use of large filter is essential for roof edges with small slope variation. The general case for S is given in (5). The corresponding SNR is given by

$$\mathrm{SNR}_{\text{general}} \approx \frac{\dfrac{h}{2} + \dfrac{|k_2 - k_1|\sigma}{\sqrt{2\pi}}}{n_o}. \tag{17}$$

2) Localization: The use of AR to detect edges does not introduce any error in the edge location in the noise-

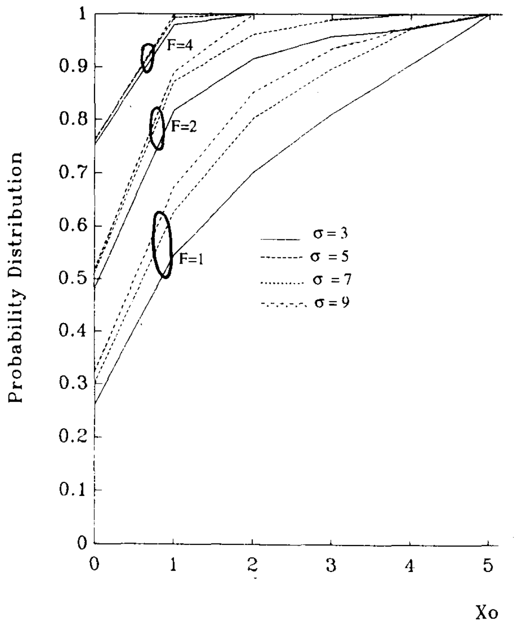


Fig. 5. Probability distribution of X_o for a roof edge with $|K_1 - K_2| = 4$.

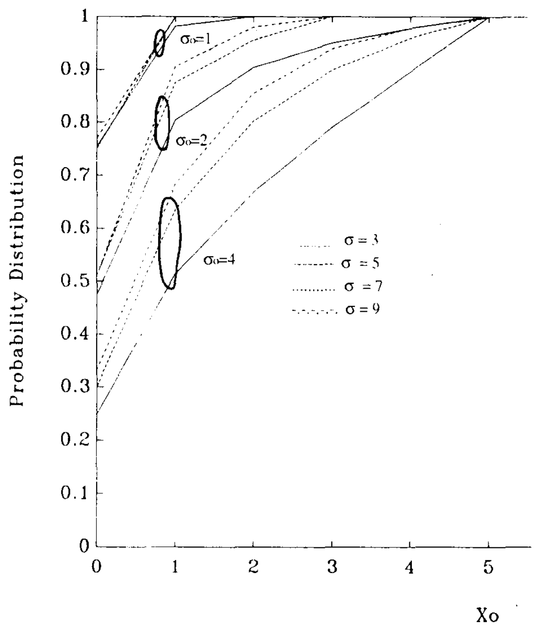


Fig. 6. Probability distribution of X_o for asymmetric step edge.

less case. This is in contrast with other approaches like LoG (see Fig. 3). In the presence of noise, the analysis becomes unmanageable. Instead of constraining the model or making unrealistic simplifications, we have performed a number of experiments to determine the localization error introduced by noise. Three synthetic images of step, roof and asymmetric step edges were tested. A varying amount of white Gaussian noise with zero mean and standard deviation σ_o was added to the images. The images were smoothed with a Gaussian filter with a standard deviation of σ and AR at each pixel was determined. A small area around the known edge point was searched for the maximum AR. The experiments were repeated 12 800 times for each image.

Fig. 5 shows the results of the experiments on a synthetic image with a roof edge. The synthetic image has a roof edge with $|k_2 - k_1| = 4$. Let us define the following figure of merit:[1]

$$F_{\text{roof}} = \frac{|k_2 - k_1|}{\sigma_o}.$$

This quantity depends on the image properties only. Gaussian noise levels of $\sigma_o = 1, 2, 4$ were added to the image. This corresponds to F of $4, 2, 1$, respectively. The image was smoothed with Gaussian filters with $\sigma = 3, 5, 7, 9$. The figure shows the estimated probability distribution of delocalization in the edge position caused by noise given that the initial edge was at $x_o = 0$. As can be seen from the figure the AR maxima was at $|x_o| \leqslant 1$ with high probability (≈ 0.98 for $\sigma = 5$) for $F = 4$. The probability can be improved with the use of larger filters. However, values of σ higher than 5 did not significantly improve the localization. The results obtained here suggests that a large scale filter is needed to detect roof edges reliably which agrees with the conclusions obtained from the SNR criteria.

The second test was performed on an asymmetric step with $h = 2$ and $|k_2 - k_1| = 4$. Fig. 6 shows the estimated probability distribution of x_o for different noise levels ($\sigma_o = 1, 2, 4$) and different smoothing filter size ($\sigma = 3, 5, 7, 9$). The results obtained for this case are similar to those of the roof edge. The figure shows that large filter is needed to obtain better localization.

The last test was performed on a step edge with $h = 20$. The F for this image is defined as

$$F_{\text{step}} = h / \sigma_o.$$

Noise levels with $\sigma_o = 2, 4, 10$ were added to the step edge. This corresponds to values of F equal to 10, 5, and 2, respectively. Four filters were used with $\sigma = 0.5, 1, 2, 3$. Fig. 7 shows the estimated probability distribution of x_o for different F and σ. As can be seen from the figure small scale filters produced better localization in general. The delocalization for the smallest filter was $|x_o| \leqslant 1$ with a 0.91 probability for $F = 5$. The results obtained here agrees with the conclusion obtained using the first criteria which suggests that a small scale filter is sufficient to detect step edges.

[1] We note the similarity between F_i and output SNR_i ((15)–(17)). The difference is that F_i depends on the properties of the image (both signal and noise), while SNR_i also depends on the filter properties.

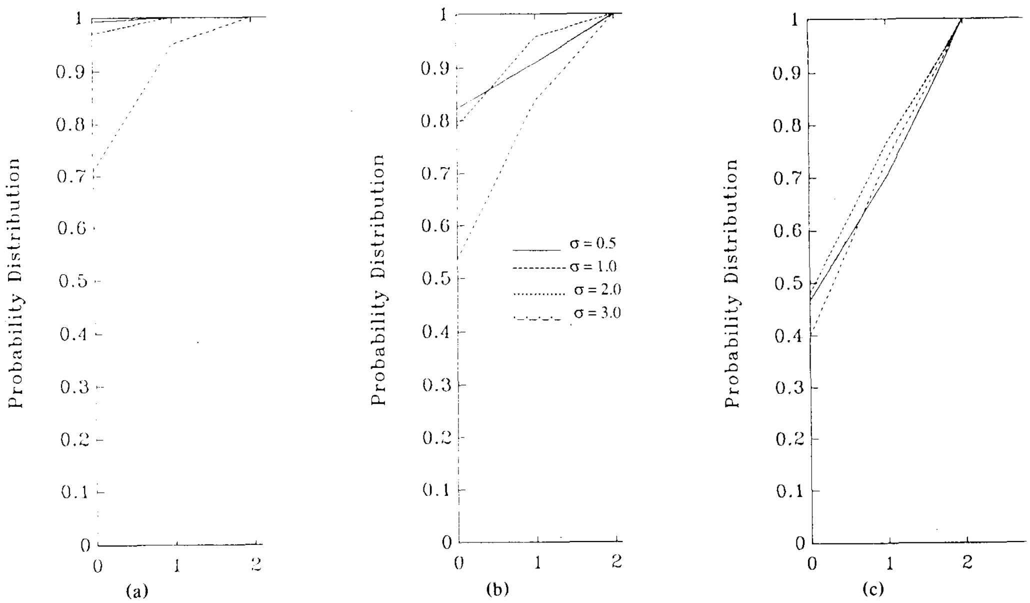


Fig. 7. Probability distribution of X_o for a step edge with (a) $F = 10$. (b) $F = 5$. (c) $F = 2$.

V. Algorithm Description and Implementation Details

The AR study given in the previous section suggests that a two stage process is needed to detect edges in range images. The first stage uses a small scale filter to detect step edges, while the second stage uses a large scale filter to detect roof edges. However, the use of large scale filters increases the probability of having multiple edges in the window. To avoid this situation, in the algorithm described here step edges are detected first using a small scale filter, thereafter, step edge points are excluded from the AR computation for roof edges. Roof edges are, in turn, detected using a large scale filter in two steps. In the first step roof edges are detected using a large scale filter and a high threshold. The second step of the roof edge detection involves computing AR using a large scale filter while excluding roof edges detected in the previous step. A lower threshold was used in this step. This allows us to detect neighboring roof edges and roof edges close to a corner more reliably.

The AR analysis presented in the previous section was done in 1-D. For the 2-D case, it is necessary to determine the direction of the edge in order to apply the methodology presented earlier. We have considered the use of surface normals to estimate the edge direction but these estimates are unreliable in the presence of noise. For this reason in our algorithm we have treated the problem as 1-D by processing the image in a number of directions around each point. For a 3×3 window AR was estimated in four directions (see Fig. 8).

The edge points obtained using the AR are usually not complete and a linking step is needed to obtain close connected contours. The resulting connected components are then determined and an iterative region growing technique based on the least square fit is used to obtain the final segmented image.

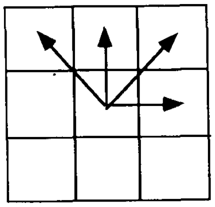


Fig. 8. Four search directions.

In summary the proposed algorithm can be divided into four steps:

1) Detect step edges using a small scale filter.
2) Detect roof edges using a large scale filter in two steps.
   a) Use a high threshold and exclude step edges obtained in the previous step from AR Computation.
   b) Use a lower threshold and exclude both step and roof edges detected in step 2a, from the computation of AR.
3) The edge contours are linked and the connected components are obtained.
4) An iterative region growing technique is used to obtain the final segmented image.

The details of the algorithm implementation are given below:

1) Detection of Step Edges: Step edges are detected as follows: The performance analysis for a step edge indicates that SNR will not improve by increasing the size of

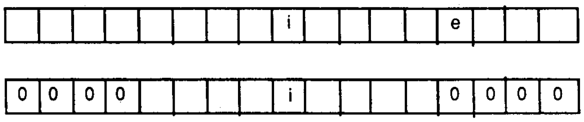


Fig. 9. 1-D Smoothing neighborhood.

the filter. Also Fig. 7 shows that the localization will not improve by increasing the filter size above $\sigma = 1$. Thus in our step edge detection we used a Gaussian filter with $\sigma = 1$ (window size = 5) to smooth the image. AR is then calculated at each pixel in four directions (Fig. 8). The point is marked as an edge if it is higher than a threshold and maxima in any one of the four directions. The threshold is not critical since the points with small AR correspond to step edges with small depth change and will be detected in the next step.

2) Detection of Roof Edges: Roof edges introduce AR values that depend on the size of the filter and the slope of the two lines forming the edge (9). The use of a large scale filter is essential to improve the detection of edges in the presence of noise. At the proximity of a corner the smoothing window cannot be extended as large as we want since step edge locations are used to restrict the AR computation. Also in the presence of two neighboring roof edges AR computation will not be reliable. Thus the detection of roof edges are divided into two steps as follows:

a) Both SNR and the localization performance of AR for roof edges (Fig. 5) improves as the size of the filter increases. However, the localization performance does not improve by much when $\sigma > 5$. We choose $\sigma = 4$ as a compromise to reduce computing requirements and to enhance the capability of detecting neighboring roof edges. AR is determined in four 1-D directions using a 17×1 Gaussian mask. Step edges obtained in step 1 are used to restrict the neighborhood size of the smoothing filter. This is a form of adaptive filtering that prevents smoothing around step edges. For each 1-D mask a search is performed to determine if an edge is present in the mask. If a step edge is detected, the value of the smoothing mask is set to zero for that pixel and for the rest of the mask (see Fig. 9).

A search in a small area in each direction is then performed. If AR at the pixel under consideration is a maximum and higher than a chosen threshold, in any of the four directions, it is marked as a roof edge.

b) The step edges detected in step 1 and the roof edges detected in step 2a are used to restrict the neighborhood size of the smoothing filter computation of AR. Similar to 2a, a small area in each one of the four directions around a pixel is searched and if the value of AR is maximum and higher than a threshold, in any one of the four directions, it is marked as a roof edge. The threshold used here is smaller than the one used in step 2a. This allows us to detect edges that have not been smoothed with a large filter (e.g., close to a corner).

3) Linking and Forming the Connected Components: Step edges detected in step 1 usually produce a thick contour. Also the roof edge detection step is allowed to accept all points that are maximum in any direction and above a certain threshold. This introduces some extra edge points. On the other hand, the roof edge curves detected might have some gaps due to factors like the presence of noise, small slope of the lines forming the edge or proximity to a corner. The resulting edge map must be refined to obtain the final segmented image. The edge points are first linked. The linking operation we used allowed us to close a one pixel gap in any one of the four directions. This operation is applied once to our test images and resulted in closed edge contours. The connected components are then obtained and labeled using a 4-connected component algorithm. The success of any region growing algorithm relies heavily on the initial seed regions. The connected components obtained in our case are large and well inside the object faces.

4) Iterative Region Growing: An iterative least square region growing approach is then used to postprocess the edge map. A large number of object faces can be well approximated by a quadratic function. Each connected component is then fitted with a quadratic surface. Let us define e_A the average error of the fit by

$$e_A = \frac{\text{total error of the fit in the component}}{\text{number of points in the component}}.$$

The pixels classified as an edge are then tested. If the least square error of adding this pixel to a neighboring surface is less than $10e_A$, the label of the point is changed to match the neighboring surface label. One pixel thick edges are excluded from the computation (i.e., no merging is allowed). Only one pass of the previous operation was needed to obtain the results shown in the next section. The quadratic surface fit is then repeated to obtain the final surface parameters. These parameters can be used to classify the object surfaces into a number of primitives.

VI. Segmentation Results

The segmentation algorithm presented in the previous section has been applied successfully on a number of range images of industrial parts. The range images were obtained using a laser range finder developed by Environmental Research Institute of Michigan (ERIM). The images are 128×128 with 8 bits/pixel. The following set of parameters were used for all the test images.

1) A 5×1 window was used for step edge detection. The threshold for accepting a step edge was set at 6.
2) For roof edges the size of the 1-D smoothing window was 17. The threshold for accepting roof edges was set at 6 for step 2a and reduced to 4 for step 2b.
3) The images are linked and the connected components smaller than 20 pixels are discarded.
4) The least square error for classifying a pixel is set at $10e_A$ for the neighboring surface.

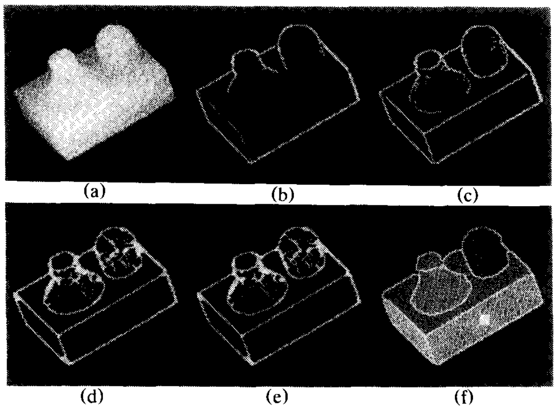


Fig. 10. Segmentation results for object_1 range image. (a) Original image. (b) Jump edges. (c) Roof edges. (d) Jump and roof edges. (e) Linked edge map. (f) Final segmented image.

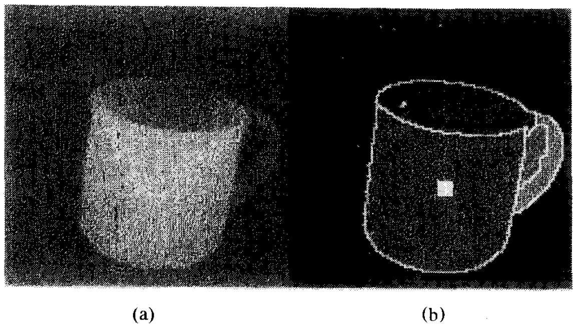


Fig. 11. Segmentation results for the coffee cup. (a) Original image. (b) Final segmented image.

Fig. 10 shows the results after each step of the algorithm when applied to an industrial part range image (object_1). The image contains step, roof and smooth edges. Parts (b) and (c) of the figure shows the edges detected by steps 1 and 2 of the algorithm respectively. As can be seen all step and roof edges in the image were detected. The smooth edges in the image are not detected. Fig. 10(d) of the figure shows the final edge map. The figure also shows the connected components and the final segmented image in (e) and (f) of the figure, respectively. As can be seen step 4 of the algorithm removed the additional edge points in the edge map.

The second image is that of a cup that has been used elsewhere [6], [13] (see Fig. 11). The noise estimated for this image is $\sigma_o \approx 1$. The final segmented image is also shown in the figure. As can be noted the handle is correctly segmented form the main body of the cup. In this respect our results are better than those reported in [6] for the same noise level. Also, the hole in cup is segmented correctly. A small region at the top of the cup is not classified (marked as edge) using the region growing algorithm. Most of the cup edges were detected in the first step (step edges) with the exception of a small part in the transition from the handle to the background (from the inside).

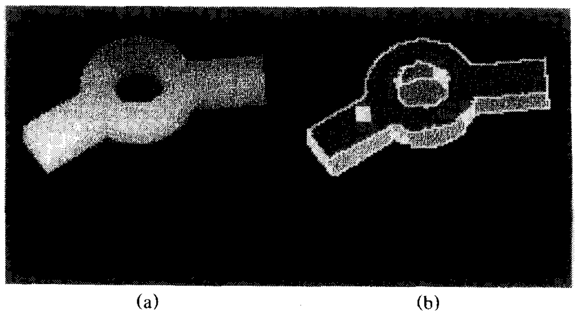


Fig. 12. Segmentation results for object 2. (a) Original image. (b) Final segmented image.

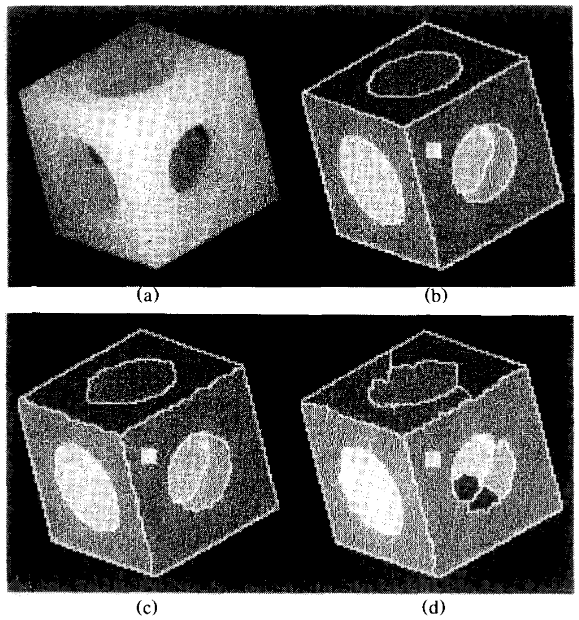


Fig. 13. Segmentation results for the cube range image. (a) Original image. (b) Final segmented image. (c) Final segmented image with Gaussian noise of $\sigma_0 = 1$. (d) Final segmented image with Gaussian noise of $\sigma_0 = 2$.

The result of the segmentation algorithm on a second industrial part (object_2) is shown in Fig. 12. This algorithm performs well for this image. A small planar region on the left side of the image was not classified. This is because the smoothing windows used are large compared to the surface size.

Fig. 13 shows a range image of a cube with through holes drilled in all the surfaces. The segmented image is shown in Fig. 13(b). The algorithm segmented the image well with the exception of two small background components that were left unclassified. To show the performance of our algorithm in the presence of noise, we added a Gaussian noise of $\sigma_o = 1,2$ to the cube image.

The segmentation results of the algorithm for these noisy images are shown in Fig. 13(c) and (d), respectively. As can be seen the algorithm performed well for these noisy images. These levels of noise are considered moderate to high for range images [20].

VII. Determining Holdsites From the Segmentation Output

The segmentation algorithm presented in the paper produces a surface-based segmentation that can be used in a variety of applications. The surfaces obtained for each object can be classified into a number of primitives (e.g., planar or curved). Surface representation can then be generated by defining relations between the classified surfaces. This representation can be used in object recognition or pose estimation by comparing them with a stored model [5], [11].

In this section we explore the use of the segmentation output to solve the problems of determining good grip points of a part for robot bin picking application. This task requires locating the object to be gripped. In the case of overlapping objects, the top-most object in the pile is usually selected [29]. The top-most object is then segmented and holdsites location, orientation and elevation is determined. This information is then supplied to the robot to plan the pickup motion. The use of range data simplifies the task considerably.

The choice of the holdsite size and shape depend on the objects under consideration and the robot hand. Tella *et al.* [28] designed several general purpose hands based on a number of guidelines for robots. A surface-adapting vacuum gripper (SAVG) was designed to handle objects possessing single surface holdsite. A contour-adapting gripper (CAVG) was designed to address problems associated with SAVG. These problems are: 1) Some objects possess surface contour too irregular for a single vacuum cup to grip. 2) It is difficult to rigidly support large objects with a single vacuum cup. 3) Single cup gripping supplies little information about an acquired object. The CAVG is equipped with a rectangular prismatic array of 20 vacuum gripper units. We have chosen this general purpose hand based on the objects we tested and the output of the segmentation algorithm.

The CAVG requires that the object should have an upper surface with a small variation in surface normal. In addition, the edge of the hand must not overlap either the periphery of the component or the edge of any internal holes when applied to the surface of the part.

Based on the previous discussion we have implemented a procedure to determine the holdsite for the objects shown in the previous section. The test images consist of isolated objects but we expect that similar procedure can be used for overlapping objects by identifying the top-most object. The output of the segmentation algorithm is used to determine the highest surface in the object. This is achieved simply by comparing the range value of the segmented surfaces. A small area (7×7) in the highest surface is then determined. The standard deviation of the AR computed[2] using a 5×5 Gaussian mask is evaluated. If the standard deviation of the AR is small (i.e., small variation in the range value) the selected area is accepted, otherwise a search for another areas is performed. This area should be inside a single surface as far as possible from the periphery of the object and any internal holes. Also, to minimize the turning moment due to the object weight, the area should be as close as possible to the center of gravity of the object. Following [28] we use the center of the area of the image as a good approximation to the center of gravity.

Figs. 10–13 show the selected holdsites superimposed on the segmentation output (small square white area). These holdsites were obtained from the final segmentation output but they could have been obtained from the edge map (steps *A* and *B* of the algorithm). The robot hand is assumed to be in the *Z*-direction (out of the picture). The position, elevation, and orientation of the holdsites is then supplied to the robot to grip the object.

VIII. Conclusion

This paper presented a new segmentation algorithm technique. The method extracts a set of features, representing step and roof edges using AR analysis. These features are then combined in an iterative region growing algorithm to obtain the final segmented range image. The results consist of quadratic surface parameters for each region and the edge type separating them. This can be used to obtain a surface representation that can be used in object recognition or pose estimation. The output of the segmentation algorithm is then used to determine potential holdsites for gripping the object by a robot equipped with a vacuum gripper. The performance of the algorithm on a number of range images is presented. The algorithm produced good results in the presence of noise.

SNR and the localization performance of the AR shows that the use of this approach is very effective in detecting edges in range data. The performance analysis shows, that for a step edge, the SNR is independent of the filter size while the localization performance improves with decreasing filter size. For a roof edge, the performance analysis shows that the SNR and localization performance can be increased by using a large filter. This indicates that this approach is especially effective for detecting roof edges since both performance criteria can be improved by increasing the filter size.

There are a number of points that we are studying to improve the algorithm. In particular, in our experiments a simple linking operation was sufficient to form a close connected contours. However, to improve the robustness

[2] For 2-D AR is defined as a direct extension of the definition in (3) i.e., $r(x,y) = |z_S(x,y) - z(x,y)|$.

of the algorithm a test on the quality of the fit in each component should be performed. If the test fails the component should be divided into two or more component. Also we are investigating the effects of using other types of filters to compute the AR.

We are also studying the use of this segmentation algorithm to generate a representation for a variety of applications. Other techniques for detecting holdsites are being considered. A possible procedure is to first extract holdsite configuration from a model, then the observed object identity and orientation is determined. Finally, the holdsites ares selected based on the object type and orientation.

Appendix

In this appendix we prove that the value of the square bracket (in (4)) is positive and has a maxima of 1 at $x = 0$. Similar prove can be obtained for the square bracket term in (6).

Lemma: If $q(x)$ is defined by

$$q(x) = \exp \frac{-x^2}{2\pi^2} - \frac{x\sqrt{2\pi}}{\sigma} \int_x^\infty G(t)\,dt$$

then for $x \geqslant 0$, $0 \leqslant q(x) \leqslant 1$.

Proof: $q(x)$ at the end points is given by

$$q(0) = 1, \qquad q(\infty) = 0.$$

Taking the derivative of $q(x)$ with respect to x we have

$$q'(x) = \frac{-x}{\sigma^2} \exp \frac{-x^2}{2\sigma^2} - \frac{\sqrt{2\pi}}{\sigma} \int_x^\infty G(t)\,dt$$
$$+ \frac{x}{\sigma^2} \exp \frac{-x^2}{2\sigma^2} = -\frac{\sqrt{2\pi}}{\sigma} \int_x^\infty G(t)\,dt.$$

Since the value of the integral is positive, the value of $q'(x)$ is negative. This means that $q(x)$ is a monotonically decreasing function. Hence $0 \leqslant q(x) \leqslant 1$ for $x \geqslant 0$.

References

[1] G. J. Agin and T. O. Binford, "Computer description of curved objects," in *Proc. 3rd Int. Joint Conf. Artificial Intelligence*, 1973, pp. 629–640.

[2] E. Al-Hujazi and A. Sood, "Range data description based on multiple characteristics," in *Proc. Goddard Conf. Space Appl. Artificial Intell.*, Greenbelt, MD, 1988, pp. 295–309.

[3] E. Al-Hujazi and A. Sood, "Segmentation of range images," in *Proc. SPIE Conf. Appl. Artificial Intell.*, Orlando, FL, 1989, pp. 229–238.

[4] H. Asada and M. Brady, "The curvature primal sketch," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 1, pp. 2–14, Jan. 1986.

[5] P. Besl and R. C. Jain, "Three-dimensional object recognition," *Computing Surveys*, vol. 17, pp. 75–145, 1985.

[6] P. Besl and R. C. Jain, "Segmentation through variable order surface fitting," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, no. 2, pp. 167–192, Mar. 1988.

[7] B. Bhanu, "Representation and shape matching of 3-D objects," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, no. 3, pp. 340–350, May 1986.

[8] R. C. Bolles, P. Horand, and M. J. Hannah, "3DPO: A three-dimensional parts orientation system," in *Proc. Int. Symp. Robotic Res.*, pp. 413–424, 1984.

[9] M. Brady, J. Ponce, A. Yuille, and H. Asada, "Describing surfaces," *Comput. Vision, Graphics, Image Processing*, vol. 32, pp. 1-28, 1985.

[10] J. Canny, "A Computational approach to edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[11] R. T. Chin and C. R. Dyer, "Model-based recognition in robot vision," *Computing Surveys*, vol. 18, no. 1, pp. 67–108, 1986.

[12] J. D. Dessimoz, J. R. Birk, R. B. Kelley, H. Martins and C. Lin, "Matched filters for bin picking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, no. 6, pp. 686–696, Nov. 1984.

[13] T. G. Fan, G. Medioni and R. Nevatia, "Description of surfaces from range data using curvature properties," in *Proc. Computer Vision and Pattern Recognition Conf.*, IEEE Comput. Soc., pp. 86–91, 1986.

[14] F. P. Ferrie and M. D. Levine, "Deriving coarse 3-D models of objects," in *Proc. Computer Vision and Pattern Recognition Conf.*, IEEE Comput. Soc., 1988, pp. 345–353.

[15] W. E. L. Grimson and T. Pavlidis, "Discontinuity detection for visual surface reconstruction," *Comput. Vision, Graphics, Image Processing*, vol. 30, pp. 316–330, 1985.

[16] M. Hebert and T. Kanada, "The 3-D profile method for object recognition," in *Proc. Computer Vision and Pattern Recognition Conf.*, IEEE Comput. Soc., pp. 458–463, 1985.

[17] M. Hebert and J. Ponce, "A new method for segmenting 3-D scenes into primitives," in *Proc. 6th Int. Conf. Pattern Recognition*, 1982, pp. 836–838.

[18] M. Hebert, T. Kanade and I. Kweon, "3-D vision techniques for autonomous vehicles," Carnegie Mellon Univ., The Robotics Institute, Rep. CMU-RI-TR-88-12.

[19] T. C. Henderson, "Efficient 3-D object representation for industrial vision systems," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, no. 6, pp. 609–617, Nov. 1983.

[20] R. Hoffman and A. K. Jain, "Segmentation and classification of range images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, no. 5, pp. 608–620, Sept. 1987.

[21] K. Ikeuchi, H. Nishihara, B. K. P. Horn, P. Sobalvarro, and S. Nagata, "Determining grasp configuration using photometric stereo and the PRISM binocular stereo system," in *Int. J. Robotics Res.*, vol. 5, no. 1, 1986.

[22] S. Inokuchi, T. Nita, F. Matsuday and Y. Sakurai, "A three-dimensional edge-region operator for range pictures," in *Proc. 6th Int. Conf. Pattern Recognition*, 1982, pp. 918–920.

[23] A. K. Jain and R. Hoffman, "Evidence-based Recognition of 3-D objects," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, no. 6, pp. 783–801, Nov. 1988.

[24] D. Lee and T. Pavlidis, "Edge detection through residual analysis," in *Proc. Computer Vision and Pattern Recognition Conf.*, IEEE Comput. Soc. 1988, pp. 215–222.

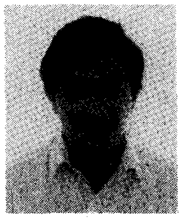
[25] W. N. Martin and J. K. Aggarwal, "Volumetric description of objects from multiple views," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, no. 2, pp. 150–158, Mar. 1985.

[26] D. Nitzan, "Three-dimensional vision structure for robot application," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, no. 3, pp. 291–309, May 1988.

[27] M. Oshima and Y. Shiria, "Object recognition using three-dimensional information," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, no. 4, pp. 353–361, July 1983.

[28] R. Tella, J. R. Birk and R. B. Kelley, "General purpose hands for bin picking robots," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-12, pp. 828–837, Nov./Dec. 1982.

[29] H. S. Yang and A. C. Kak, "Determination of the identity, position and orientation of the topmost object in a pile," *Comput. Vision, Graphics, Image Processing*, vol. 36, pp. 229–255, 1986.



Ezzet Al-Hujazi (S'90) was born in Baghdad, Iraq, 1957. He received the B.S. and M.S. degrees in electrical engineering from the University of Baghdad, Iraq, in 1977 and 1979, respectively. He is currently a Ph.D. student at Wayne State University, Detroit, MI.

He was a faculty member at the University of Technology, Baghdad, Iraq. Currently he is a visiting research associate at George Mason University, Fairfax, VA. His interests include computer vision, computer graphics, and image processing.

Arun Sood (S'67–M'67–M'81–SM'83), for a photograph and biography, please see page 1242 of this Transactions.