BIFOCAL VISION: A Holdsite-Based Approach to the Acquisition of Randomly Stacked Parts

Daniel Kornitzer

B Eng Génie Electrique, Ecole Polytechnique, 1985

Department of Electrical Engineering

McGill University

A thesis submitted to the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

M. Eng., McGill University, 1988

February 15,1988

Abstract

The problem of binpicking consists of computing sufficient information about identity, position and orientation of 3-D objects randomly stacked in a bin, in order to allow a robot to individually grasp a part and place it at a specified pose.

In this thesis we describe the BIFOCAL VISION system which we have developed to enable a PUMA 260 robot to grasp and place industrial parts which are randomly piled and oriented in a bin. This is achieved through the graceful integration, as visual feedback signals, of the sensory inputs provided by a 2-D television camera positioned over the workspace and a wrist-mounted single-point range finder.

The standard approach is to first attempt the recognition of identity and pose of the part and then send the robot hand to an appropriate holdsite so that the part can be grasped and moved The main disadvantage of this method is that it usually is very difficult to recognize a heavily occluded part in a 2-D image

The BIFOCAL approach integrates information from two types of sensors. First, visual information from a camera is analyzed to isolate the location of potential holdsites in the 2-D image. The robot gripper is then sent to the most promising holdsite using line-of-sight control. Second, close to the object the robot is guided by a single-point range finder and acquisition is attempted.

The complete system has been developed and tested on different types of objects such as cylinders, rings, bolts, etc. We have also evaluated the system's sensitivity to variations in scene lighting, holdsite models and the amount of collected range data. On the basis of these experimental results, we have found that the BIFOCAL VISION system provides robust and reliable binpicking.

Resumé

Le problème de l'acquisition robotisée de pièces empilées au hasard consiste à déterminer l'identité, l'emplacement et l'orientation des pièces, de façon à permettre au robot de les saisir et de les manipuler une par une.

Dans cette thèse nous décrivons le système BIFOCAL VISION que nous avons développé en vue de doter un robot PUMA 260 de la capacité de saisir et placer des objets tri-dimensionnels empilés au hasard, et ce grâce à une intégration adéquate des signaux fournis par deux senseurs visuels: une caméra TV placée au-dessus de la pile d'objets et un senseur ponctuel de profondeur installé sur la main du robot.

L'approche conventionnelle consiste à déterminer d'abord l'identité et l'emplacement d'une pièce, et ensuite à envoyer le robot afin d'acquérir ladite pièce. Cette technique comporte un désavantage important: il est souvent très difficile de reconnaître des pièces partiellement visibles dans une image de luminosité.

Notre approche comporte deux étapes essentielles. Le premier module utilise une image globale de la scène, fournie par une caméra TV, afin de trouver l'emplacement de points de saisie potentiels pour une pince à doigts parallèles. Le deuxième module obtient, à l'aide du senseur de profondeur, une grille de données tri-dimensionnelles autour du meilleur point de saisie potentiel,

de façon à confirmer la présence du point de saisie et à calculer son emplacement précis et son accessibilité au moyen de la pince du robot. Ensuite le robot prend la pièce et la dépose à l'endroit voulu et avec l'orientation désirée.

Le système BIFOCAL VISION a été vérifié en utilisant différents types de pièces, telles que des cylindres, tores, vis, etc. La sensibilité du système par rapport à l'illumination, les modèles des points de saisie et la quantité de données tri-dimensionnelles échantillonnées a aussi été évaluée. Les résultats obtenus nous permettent d'affirmer que BIFOCAL VISION effectue l'acquisition de pièces empilées au hasard de façon robuste et fiable.

Acknowledgements

I would like to thank my advisor, Dr. Martin D Levine, for his continuous support and interest, and for his many ideas and suggestions without which this thesis would not have been possible.

I would like to thank the members of the staff and my fellow graduate students whose help throughout my research work has been greatly appreciated, in particular John Lloyd (Mr. RCCL), Bruno Blais, Benjamin Kimia, Mike Parker, Abdol Mansouri, Fréderic Leymarie, Gregory Carayannis, Guy Godin, Dominic Chau and Adrian Zimmermann.

I would also like to thank my parents and sister for their encouragement and patience. Finally, I would like to acknowledge the financial support provided by NSERC.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

The problem of feeding workpieces that are unoriented in bins is an ubiquitous one in manufacturing. Solving this problem should prove useful for the numerous applications that require robots to acquire and manipulate objects whose orientations are unknown. The purpose of most of the existing algorithms is to use sensory data, mostly visual, so as to recognize the identity, position and orientation of the parts. A robot could then be able to feed the oriented parts to an automated assembly line. Thus, the problem of binpicking consists of computing sufficient information about the identity and pose (i. e., position and orientation) of 3-D objects randomly stacked in a bin, in order to allow a robot to individually grasp a part and place it at a specified pose. This process may be repeated for every part in the bin.

The standard approach is to first attempt the recognition of type and pose of the occluded part, and then send the robot manipulator to an appropriate holdsite so that the part can be grasped and moved. The main disadvantage of this method is that it usually is very difficult to recognize a heavily occluded 3-D part in a 2-D image. Virtually every published algorithm in this class deals only with two-dimensional flat objects. Hence, it is reasonable to assume that the bin-of-parts problem has been solved for the case of

flat objects and, to a certain extent, of 3-D parts with a few stable positions even when piled in a bin. Lowe [32,33] constitutes an exception, since he claims that the viewpoint consistency constraint can lead to robust three-dimensional object recognition from single gray-level images. Another approach consists of first using computer vision techniques to isolate the location of potential holdsites in an image. The robot gripper is then sent to the most promising holdsite, and an attempt is made to grasp the unknown object. Upon successful grasping, the part's identity and pose are more easily computed. This method reduces the complexity of the initial problem by breaking it into two which are simpler to handle Current implementations of this approach do not seem to be very reliable, nor do they take full advantage of the wide variety of sensory devices available, such as tactile and range sensors

The proposed approach is holdsite-based It uses a CCD TV camera and a single-point range finder as sensors The visual input is used for holdsite detection and fast control of the manipulator, whereas the depth data provides close-in control This results in a reliable and robust system for part acquisition and manipulation.

This thesis is organized as follows. the next chapter consists of a survey of previous work in the area of binpicking and related topics Then, the physical components of the system are described in chapter 3, whereas chapters 4 and 5 describe the 2-D and 3-D image processing algorithms used in this project. The experimental results are shown and subsequently discussed in chapter 6 Finally, in the last chapter, we present the conclusion

Chapter 2 Survey

This chapter essentially consists of a survey of a large number of papers related to the bin-of-parts problem They basically deal with the recognition of the identity, position and orientation of overlapping parts randomly piled in a bin The literature is covered in a structured manner Methods are classified by, among others, the type of features used for recognition, the sources of sensory data, and the methodology Finally, the state-of-the-art in binpicking is described

2.1 Classification Criteria

The classification of methods used for the recognition of industrial parts is not a trivial task. The literature is rich in this subject, so it is necessary to group methods on the basis of certain key parameters. Several of these have been selected, although some parameters are not completely independent of the others. These are listed as follows:

(i) Source of sensory data magnetic sensors, tactile sensors, range scanners, television cameras or a combination of two or more sensors.

(ii) Type of features used for recognition: local, global or both.

(iii) Type of objects that the system is able to recognize: mostly flat 2-D objects or 3-D objects.

(iv) Arrangement of objects in the bin: in the simplest case, each object to be recognized must be completely visible and surrounded by background. In a more complex case, objects are allowed to touch neighboring ones but not overlap. In the most general case, objects are allowed to touch or partially occlude one another.

(v) Methodology to detect identity and pose *before* grasping by a robot or to use a holdsite-based approach in which legal grasp configurations are first detected, and once the object is held by the robot's gripper, the identity and pose are computed.

(vi) Type of matching algorithm. data-driven, model-driven or a mixture of both

(vii) Method of entering object models into the computer manually, using a teach-by-showing technique or retrieving models from a CAD/CAM database.

Among the above parameters, the type of the matching algorithm is one of the most complex to determine. Usually it is composed of a mixture of both data and model-driven modules. However, often one of the components clearly dominates the other, and thus it is relatively difficult to assign a type to the algorithm. This is a basic control issue, namely, whether recognition is triggered by high-level expectations or by low-level visual input. In its pure form data-driven processing is also known as bottom-up control. The image is first preprocessed in a domain-independent way; then it is segmented into meaningful regions or contours, and finally the objects and their relations are identified.

In a similar way, model-driven processing is known as top-down control. Predictions are generated by internal models in the knowledge database, and then the verification of these predictions leads to image understanding. This goal-oriented paradigm is also called *"hypothesize and verify"*. It seems clear that neither of the the two matching types in its pure form is well suited to computer vision. This fact has brought about some very interesting combinations of the two matching algorithm types.

2.2 Binpicking Algorithms

2.2.1 Blind Acquisition

The simplest way of solving the bin-of-parts problem is to use a blind robot equipped with only local sensing devices. It could acquire pieces by physically scanning a bin until contact with a piece was sensed, and then use tactile sensing capabilities to pick up the piece. Blind acquisition systems have been implemented using magnetic, vacuum or one-fingered hands [15,41]. This technique has several drawbacks: an inherently low probability of finding workpieces along the search path due to its blindness, and the long time constants associated with arm motions. It may also be impossible to design grippers which can blindly acquire all types of workpieces, magnetic techniques only work for metallic parts, and vacuum cups cannot easily pick large pieces nor those with irregular surfaces.

2.2.2 Methods Using Global Features

We will now consider robotic systems equipped with sensing capabilities, namely computer vision systems. First, we will focus our attention on algorithms that use global

features for object recognition. A global, as opposed to local feature, is one that depends on the totality of the object Typically global techniques for object recognition consist of pattern recognition using global feature vectors [25] Each object is described by a list of numerical values that are as invariant as possible with regard to translation, rotation and scaling of the object During the system's learning stage, a model (or a feature vector) is computed and stored for every possible object type In the recognition phase, the feature vector of the unknown piece is compared to the feature vectors of each object type The workpiece is recognized using a nearest neighbor or likelihood ratio classifier This procedure can be slow if the feature vector is large and time-consuming to compute, or if there is a large number of models against which it has to be matched Thus, a binary decision tree may be used to speed up recognition time [5] Beginning at the root node, a single global feature is computed Different branches of the tree are taken depending on the value of the feature Thus, all objects connected to branches other than the one chosen are eliminated from consideration. One by one, more features are computed and compared to thresholds, reducing the possible-object set, until only one object is left in the set Yachida and Tsuji [55] used a binary decision tree, in which the next feature to look for was based on the current possible-object set

Among the global features that can be used are moment invariants [14,22], as given by equation (2 1)

$$m_{p,q} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \rho(x,y) x^p y^q dx dy \qquad p,q = 0,1,2,... \tag{2.1}$$

where $\rho(x,y)$ is the density function which can be either zero or one in the case of a binary image. In the more general case of a gray level image, the density function corresponds

to image intensity. These moments are then algebraically combined to yield a sequence of moment invariants which are subsequently used as a feature vector.

The SRI vision module [17] uses a set of heuristically determined global features, such as the area of the binary silhouette of the object, perimeter, number of holes, area of holes, maximum and minimum radii from the object's centroid, the ratio of these two radii, etc. A similar approach was taken by Pugh [39] who proposed efficient algorithms for computing moments.

Another global feature set is the normalized Fourier descriptor, which has been used to recognize aircraft from silhouettes [61]. The Fourier descriptor of an object is found by taking the discrete Fourier transform of its contour The boundary curve is treated as a periodic complex function with real and imaginary parts corresponding to the x and y coordinates. The descriptor is then normalized to a standard location, rotation angle, size, and contour trace starting point. During recognition, the normalized descriptor of the unknown object is compared to each one of the models stored in the object database, and object identity is determined by the closest match.

As mentioned earlier, global features are dependent on the totality of the object, and hence they cannot be used to recognize objects that are only partially visible, such as objects that are partially in the field of view or occluded by others. The reason is that global features computed for part of an object are, in general, different from those computed for the entire object. To solve this problem many researchers have opted for the use of local features, which depend on parts of an object, and can therefore increase the possibility of finding identity and pose of overlapping objects.

2.2.3 Methods Using Local Features for the Recognition of 2-D Objects

Local features can be computed on the basis of different types of sensory input, such as two-dimensional brightness images, range maps, and even tactile data. Our review of the literature will start with the algorithms that use 2-D images in order to recognize mainly flat objects Almost every method in this group is model-driven.

Bolles and Cain [9] introduced the local-feature-focus method, which is an algorithm designed to recognize and locate occluded two-dimensional objects. The local features used are holes, convex corners, and concave corners, as shown in figure 2.1. The first step is the detection of the type, location, orientation and size of the local features found in the image. The local-feature-focus models are generated by an algorithm that performs a detailed analysis of computer aided design (CAD) models of the objects and searches for a cluster of local features in a relative configuration that does not occur elsewhere in the same object nor in any other object in the database. One feature in this cluster is selected as the "focus" feature. The second step is to search for objects in the image. This is done by sequentially searching for their focus features. When one of them is found, its neighborhood is searched for the remaining features in the cluster. If these are found in a configuration consistent with the model, then the object is hypothesized to exist at the location. The system uses a maximal-clique algorithm [8] as a graph matching technique so as to locate the largest cluster of mutually consistent assignments. Finally, the object's template is translated and rotated as required, and then matched to the image. If the match yields a good result, the hypothesis is considered to be verified, and the object is recognized.

Perkins [37,38] proposed a system which can determine the position and orien-

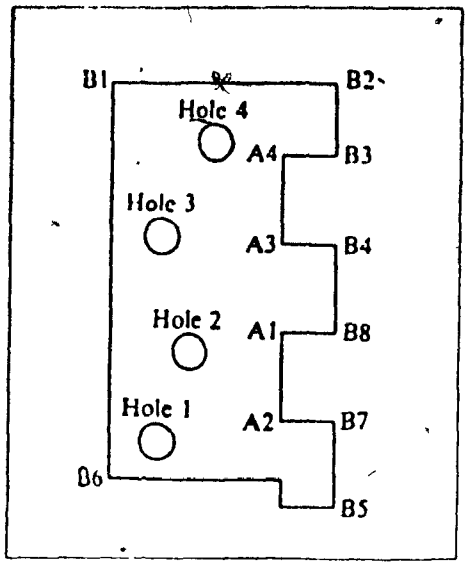


Figure 2.1 Local features of a hinge part from [9]

tation of complex curved objects in noisy gray-level scenes First, edge points are detected in the image using the Hueckel operator [23] These edge points are linked and stored as chain codes The edge chains are then approximated by straight lines and circular arcs by fitting lines, using a least-squares fit to the chain data in θ-s space (i e angle-arc length space) Thus, the system organizes and reduces image data to a compact representation having the appearance of a line drawing (see figure 2 2) This representation is used for forming object models by sequentially showing every possible object to the camera under favorable lighting and background conditions Under these conditions the system stores the detected "concurves" (i e curves) as models As the program tries to recognize objects, image curves are matched against model curves. Possible matches are suggested by the curve's type, length, total angular change, bending energy, and several other of its properties. At this point, potential matches are checked using cross-correlation in the θ-s space. Finally, if the results are adequate, matches are verified by computing a transformation from model coordinates to image coordinates and by searching for edges in the expected directions at a list of points spaced along the model's perimeter If this test

provides enough supporting evidence, the object's identity and pose are determined. An earlier system proposed by McKee and Aggarwal [35] also used similar techniques.

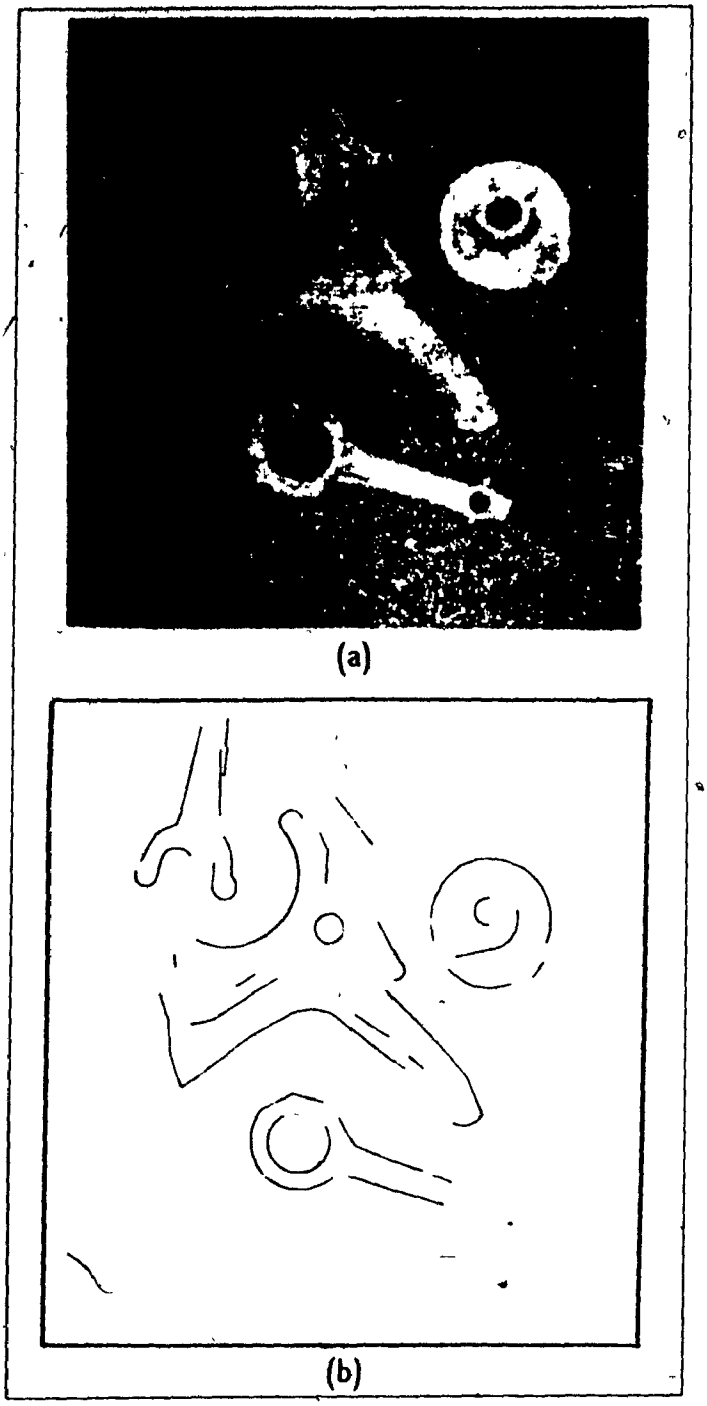


Figure 2.2 Example of concurves, adapted from [37]

Ayache and Faugeras [3] introduced HYPER (HYpotheses Predicted and Eval-

uated Recursively), a recognition method based on the generation and recursive evaluation of hypotheses Whether the system is trying to build a model or a scene description, it performs the same sequence of operations on the image If the object-background contrast is high enough, the image is thresholded into a binary image and then smoothed using erosions and dilations, which are mathematical morphology operators [47] However, under more general lighting conditions, edges are found by combining gradient (Sobel) and second order derivative information (zero crossings) At this point the program builds a list of connected border points, after which the connected components are approximated by polygons Shapes of 2-D objects are therefore represented by polygonal approximations of their boundaries The ten longest segments of the model description are sequentially matched against the segments of the scene description so as to generate hypotheses These are evaluated by attempting the identification of additional segments between the two descriptions Also, the predicted position of the model is refined by a Kalman filter The matching ends when a sufficient number of hypotheses has been tested or if a very high quality match is obtained Finally, the best hypothesis is reexamined so that it can be either validated or rejected

Knoll and Jain [28] describe a system for recognizing partially visible objects using feature indexed hypotheses Each local feature is associated with a list of where it occurs in the object models When a match is found for a feature in the image, objects are hypothesized for each object identity and pose in the feature's list These hypotheses are tested by first translating and rotating the object model to the hypothesized location, and then verifying at a periodic sampling of points along the object model boundary, that the predictions are fulfilled in the image Using this algorithm, recognition time grows only as the square root of the number of possible objects. It is worth noting that a non-optimal

procedure is provided for automatic feature selection, given a set of possible objects.

Turney et al. [49] introduced an algorithm to recognize and locate partially occluded 2-D parts using a subtemplate based version of the Hough transform [30]. The subtemplates are overlapping segments of the object model boundaries. Each subtemplate is assigned a weight which is a measure of its distinctiveness or saliency. The saliency of an object's subtemplate is entirely dependent on the set of possible objects and therefore embodies a priori knowledge about what can appear on the scene (see figure 2.3). The subtemplates are sequentially matched to the image using a least-squares fit in the θ-s space. Whenever a match is found, the accumulator pointed to by the subtemplate's vector (i e the hypothesized object's centroid) is incremented. Finally, when all the subtemplates have been matched against the edges in the image, the accumulator with the largest value is selected If it is above a certain threshold, the object is recognized, with the accumulator location indicating the object's centroid. The main advantage of this method over earlier techniques is the weighting scheme that increases the importance of the most distinguishing features found in the set of possible objects

Koch and Kashyap [29] proposed a vision system to identify occluded industrial parts. First, objects are separated from the background using a simple thresholding technique. Next, the boundaries are extracted by a contour following algorithm. At this point, the boundaries of the objects are smoothed using a polygon approximation procedure. The result is a grouping of the contour points into line segments. From the polygon approximation the curvature function of the boundary is estimated. Vertices with positive curvature are labeled convex, while those with negative curvature are labeled as concave. Corners are detected as local maxima of the absolute value of the curvature function and are used as

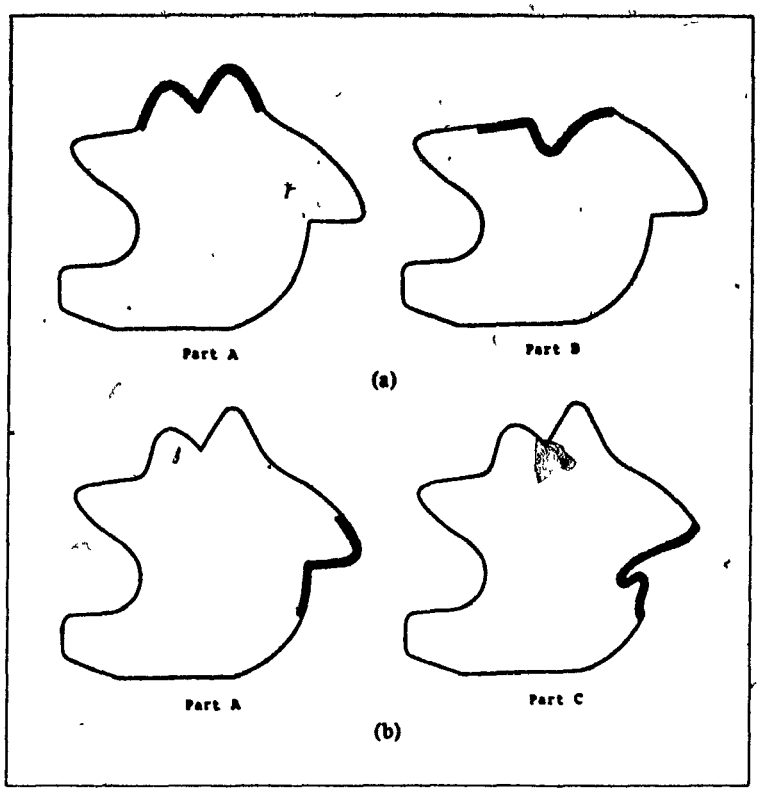


Figure 2.3 Context dependency of salient features, from [49]

local features for matching purposes. For every matching between an image feature and a model feature the corresponding coordinate transform is computed to map the model corner into the image corner. This transform is the best in a least squares sense. A group of consistent matches can easily be recognized, since they all have approximately the same coordinate transform. Therefore, hypotheses are generated by clusters of consistent corners in the image. Hypothesis verification is performed by projecting the boundary of the model onto the image, using the transform previously found, and checking the interior of the contour for consistency. Depending on the outcome of this test the object is recognized or the hypothesis rejected.

Hattich [20] proposed a strategy in which the boundary of the objects is sequentially constructed, on the basis of consistent local evidence, by a model-driven algorithm.

The algorithm is able to "jump" over occlusions and is reported to work well for two-dimensional overlapping parts.

Wallace et al. [52] describe a system that uses local shape descriptors for recognition of single aircraft silhouettes. Although not used for binpicking the method is general. First, the boundary of the object is traced, and peaks and valleys in the curvature function are detected. The local shape descriptors are arc length between peaks and angle change between valleys. The list of image features is matched against the lists of the models, and the unknown object is recognized by the closest match. Tejwani and Jones [48] used a similar system for the recognition of partial shapes.

Relaxation techniques have also been used for the purpose of matching two-dimensional shapes. Local features in the image are computed and then matched to all the features in the object models. For each image feature a vector is stored containing the estimated probability that it corresponds to each feature in the models. These vectors are updated by a relaxation algorithm in which neighboring consistent labelings support each other. At the end of the labeling process image features are recognized as their vectors' entries with the highest probability. Methods using this approach can be found in [6,11,44,45].

The Hough transform and modified versions of it have been used for shape recognition, such as the generalized Hough transform which detects arbitrary two-dimensional shapes [4] and a subtemplate based version of the Hough transform that recognizes occluded objects [4]. Segen [46] describes a Hough based technique with a particular search method. Initially, objects have three degrees of freedom, namely rotation and x-y translation. These three dimensions are reduced one at a time by the use of one-dimensional

Hough transforms.

The methods described above differ from each other mostly with regard to the local features used, and on the way in which they perform matching between models and data. The basic approach to binpicking is that of object recognition, which is in turn defined as a *representation and search* problem. In view of this paradigm, local features are the means to represent the world, whereas matching algorithms correspond to search procedures that ensure scene interpretation in a limited context.

This concludes the review of papers concerning the recognition of identity, position and orientation of two-dimensional objects.

2.2.4 Methods Using Local Features for the Recognition of 3-D Objects

With regard to binpicking three-dimensional parts we will first consider systems that use television cameras. Systems with other types of sensors, such as tactile sensors and range cameras, will then be discussed. This classification of methods, by the sensor type they use, seems to be adequate, since systems in the same group have to face similar limitations and constraints due to the nature of their sensory input.

Kelley et al. [26] at URI (University of Rhode Island) have developed three vision algorithms for binpicking. All the algorithms are holdsite driven, that is, they recognize the location of potential gripping points for a particular type of gripper. Part acquisition is then attempted, followed by the computation of object's pose. Two types of grippers are used: a vacuum cup gripper and a parallel-jaw gripper. Object grasping with the vacuum cup requires the detection of patches of smooth surfaces, whereas with the parallel-jaw gripper

it requires two opposing parallel edges, linear or curvilinear. The heuristic techniques detect these surfaces and edges, and thus they provide a strong indication of potential holdsites. One such technique is called shrinking. It is mainly used for finding planar surface patches where a vacuum cup can be applied. First, a gray level picture of the scene is taken Intensity and gradient thresholding are then applied so as to separate parts from background, and also overlapping parts from one another This last procedure outputs a binary image, which is subsequently eroded The shrinking operation amounts to iteratively peeling the boundary of the objects until a preset number of iterations is achieved, as shown in figure 2 4 The remaining pixels are clustered by distance into regions The largest of these regions are labeled as potential holdsites for a vacuum cup gripper

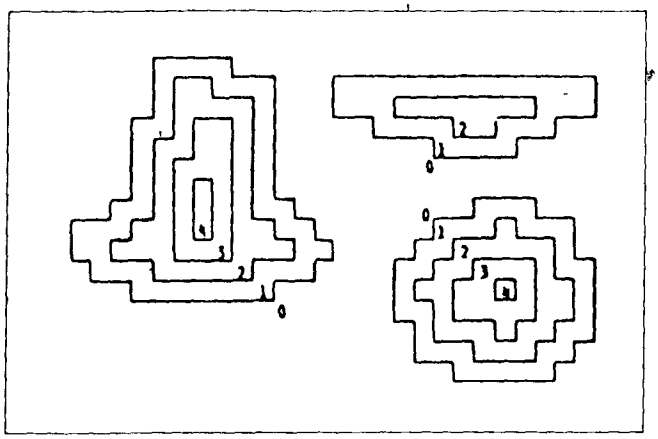
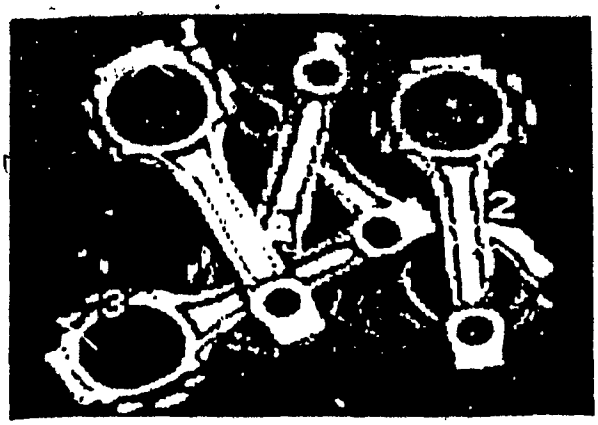
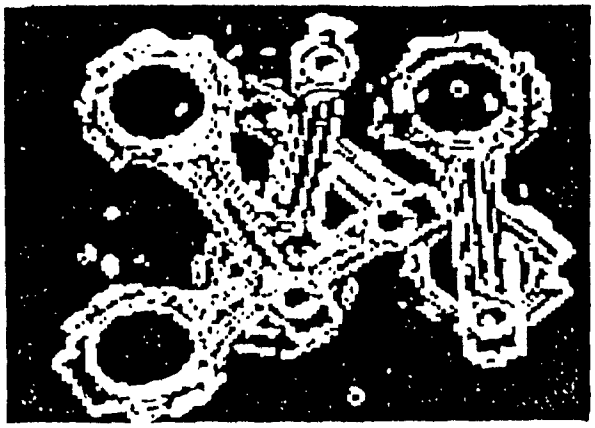


Figure 2.4 Different erosion levels as obtained by the shrinking procedure (adapted from [26])

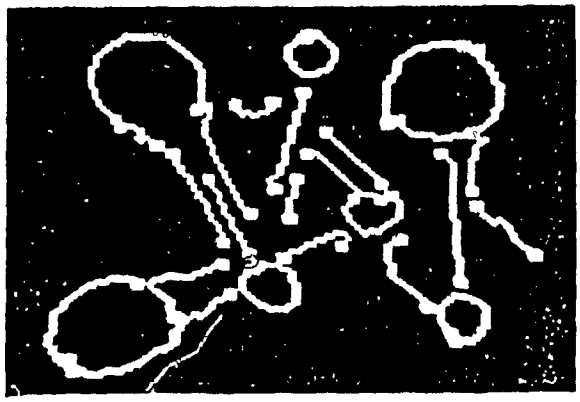
Another method is the collision fronts algorithm, which can be considered as a gray-level version of the shrinking algorithm. Its objective, however, is to search for opposing parallel edges which correspond to parts that can be grasped by a parallel-jaw gripper. This algorithm attempts to obtain the reduced skeleton, namely the subset of the skeleton that is bounded by long parallel edges, by propagating edges towards the middle of the part. Whenever a propagating edge encounters an edge being propagated from the



(a)



(b)



(c)

Figure 2.5 Collision fronts applied to connecting rods (a) shows the original image, whereas (b) illustrates the gradient image and (c) the collision fronts (from [26])

opposite direction, a collision point is formed. Collision points are subsequently clustered into collision fronts by using a line merging technique. The longest collision fronts indicate the presence of potential holdsites, as shown in figure 2.5.

The third method proposed is the parallel-jaw filter algorithm, which uses matched filters for detecting holdsites (see figure 2.6). In all, four "eigenfilters" are applied to the image, yielding the position and orientation of potential gripping points in the form of parallel clamping surfaces. The filters are rotated versions of the parallel-jaw template. The shrinking, collision fronts and parallel-jaw filter methods are complementary, since they can be used for different gripper types, objects and lighting environments. A robotic system using these algorithms has also been implemented with reportedly good results. Further work on the same holdsite-driven approach, by researchers at URI, can be found in [12,13,27].

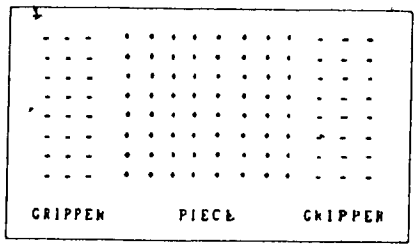


Figure 2.6 Parallel-jaw filter schematic showing an appropriate operator structure for holdsite detection Maximum response is obtained when the central region corresponds to the holdsite and the lateral ones map into free space (from [26])

Boissonnat [7] describes a method for matching a robot hand structure to an object's contour. The location of stable holdsites is computed by means of a local analysis of the object silhouette The polygonal approximation of the silhouette is segmented into primitives, and these primitives are then parametrized. The final result is the complete list of possible grasps. The main advantage of this algorithm is that it follows a well-defined general procedure for finding stable gripping points. However, it requires a properly computed object silhouette as an input.

Fukada et al. [16] built a system which recognizes crankshafts that are tightly arranged and piled up in multiple layers. Their algorithm first carries out the connectivity

analysis of the input binary image. Then it computes elementary blobs by using a line fitting procedure on boundary pixels of connected regions. At this point, blobs are matched to object components, and finally groups of blobs are recognized as objects on the basis of relational models Object pose is computed simultaneously with the recognition stage.

Horn and Ikeuchi [21] have used photometric stereo to find surface orientation at every pixel Three images of the scene are obtained using a single CCD television camera and three different light sources Triplets of intensity values for the same pixel under three different lightings are mapped into surface orientation vectors by means of a look-up table developed using a calibration object The result of the photometric stereo module is called a needle diagram of the scene, since it can be shown as a picture of the surface covered with short needles, each needle being parallel to the local normal to the surface A segmentation procedure is then carried out This procedure divides the input scene into isolated regions based on the surface orientation data generated by the photometric stereo module Edges are detected in areas where the surface normal varies discontinuously with position, and also in areas where surface orientation is undefined due to either mutual illumination or shadowing Once the image has been segmented, one region is selected on the basis of its area and Euler number Figure 2 7 shows the detailed needle diagram over the target region Next, an orientation histogram is generated for the selected region. The orientation histogram is a discrete approximation of the Extended Gaussian Image (EGI) The EGI of the object (i.e the region) is matched against model EGIs in order to determine the object's attitude. Object identity is assumed to be known, but could also be found by EGI matching. Furthermore, Ikeuchi et al [24] have added a binocular stereo module to their photometric stereo system. Binocular stereo generates range data so as to produce a coarsely sampled elevation map of the scene. This depth map is used by the planning

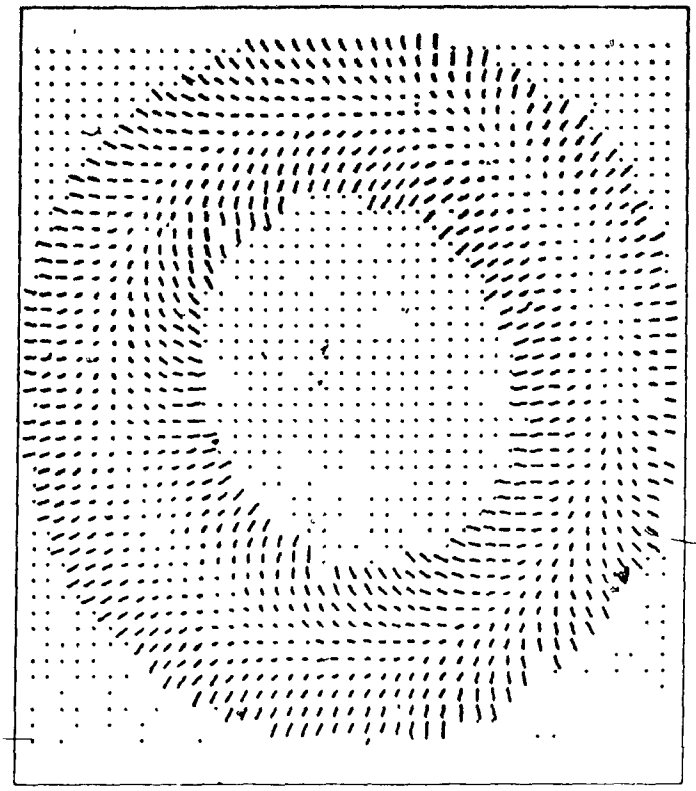


Figure 2.7 Detailed needle diagram (from [24])

process in order to compute collision free grasp configurations

Tactile sensors are also used to provide range and surface orientation data Grimson and Lozano-Perez [18] describe a system that has tactile sensing capabilities An object in the field of view can be identified by analyzing sparse range and surface normal data obtained by the sensor Object models are stored in a CAD-type database and have discrete faces Matching is performed by pairing range data points to faces in the object models

Equation 2.2 shows that for m known objects with n_j faces each, and s range data points, there are c possible combinations of pairings

$$c = \sum_{j=1}^{m} (n_j)^s \qquad (2.2)$$

The resulting tree of combinations is searched to find a consistent set of pairings, which is the basis for object recognition. The tree search may be accelerated by the use of face distance and normal constraints that prune almost all of the combinations. Grimson [19] has also proposed techniques for acquiring position and surface orientation data about points on the faces of objects so as to select sensory points that will force a unique interpretation of the identity and pose of the object with as few data points as possible

Rodger and Browse [40] have proposed a system that attempts to integrate visual and tactile inputs Object models are clearly edge-based, in that objects are broken into faces, which are in turn described by their edges Visual input is used to detect edges and to compute their length and attitude, whereas tactile input indicates the location of corners, edges or flush contacts. Matching and object recognition are performed on the basis of sensory data provided by both sources of input, namely, edges and corners. The approach seems to be adequate, however a system implementing this algorithm has yet to be built

2.2.5 3-D Vision and Binpicking

We will now turn our attention to some research efforts which attempt to solve the bin-of-parts problem by using three-dimensional vision (i.e. range maps). Yang and Kak [53,54] describe an algorithm for detecting the identity and pose of the topmost object in a pile. Objects may be planar, like those of the convex polyhedral type, or curved, such

as those that can be identified uniquely by using EGI's. The first step in the algorithm is finding the highest point in the scene, which is assumed to belong to the topmost object in the pile. Planar objects are then segmented by a region growing procedure based on surface normal adjacency and object normal constraints. Once the object is isolated, its EGI is computed so as to be able to recognize the object's identity and attitude. In the case of curved objects boundary detection is used to segment the topmost surface, its interior is then filled in. Object identity is detected by means of surface curvature analysis, while the EGI is matched with prototype EGIs in order to yield object pose.

Agin et al. [1] use local features found in three-dimensional images so as to recognize randomly oriented piled objects. Their method is called "pose cluster matching". It consists of first computing single local-feature assignments, after which the algorithm finds mutually compatible sets of features that constrain the match pose. Finally, clustering is performed. Object identity and pose are dictated by the largest set of consistent features.

Archibald and Rioux [2] have built WITNESS, a system for object recognition using range images. Planar surface extraction is performed by means of clustering, on the basis of slope, and region merging. Objects are modeled by augmented surface adjacency graphs. Thus, matching an object with a model is equivalent to graph matching for isomorphism. The method used is called heuristic augmented graph matching and takes into account the following constraints: structural compatibility, relational compatibility and reliability.

Bolles and Horaud [50] have extended the local-feature-focus method [9] in order to detect 3-D objects using range maps. Objects are recognized one by one, on the basis of clusters of consistent features, and then the system builds a global description of the

scene that describes which objects are on top of the others.

Van Laethem et al. [50] describe a holdsite-based approach to binpicking. First the range image is approximated by flat regions. Then, the algorithm searches for gripping sites that the gripper can access following collision-free trajectories. Roth [42,43] also uses a holdsite driven method. His algorithm finds the highest point in the range image and then computes the orientation of the major axis of the object containing the highest point. Several profiles of the object are subsequently collected perpendicular to its major axis, and finally the best legal holdsite is computed. This is achieved by using a set of parameters for measuring holdsite quality: slippage, which is dependent on the angle between the two clamping surfaces, stability, which is proportional to the area of contact between the gripper fingers and the object; and safety, which is related to translational uncertainty. Figure 2.8 illustrates a holdsite detected using this algorithm.

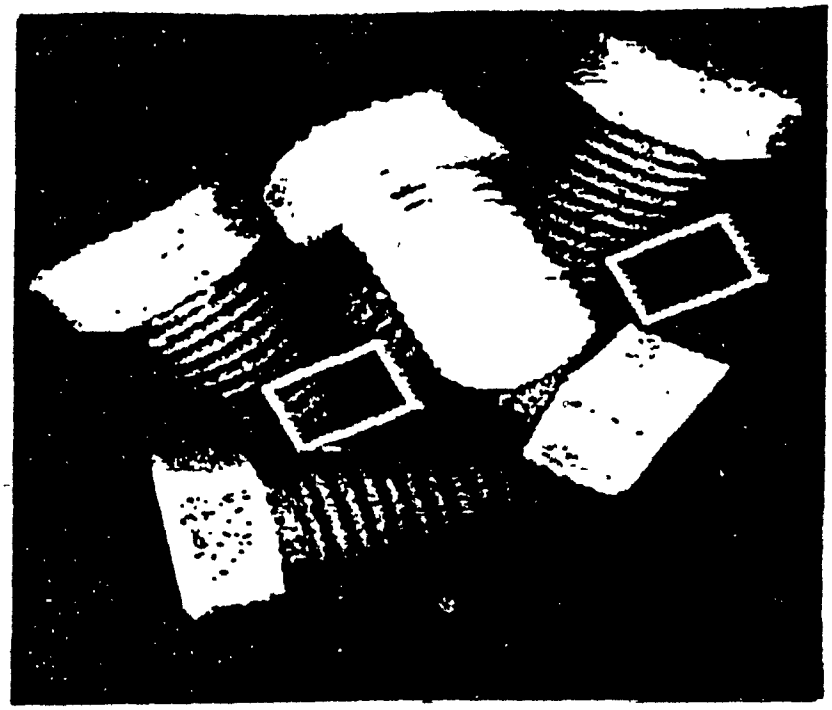


Figure 2.8 Example of a computed holdsite whose presence is indicated by the two white rectangles (from [42]).

A very interesting approach to object recognition using range images has been proposed by Oshima and Shirai [36]. Object models consist of regions, with their respective properties, and the topological relations between them. That is to say that models are graphs which have regions as nodes and topological relations as branches. Matching an image to a model is a combination of data-driven and model-driven search processes, as shown in figure 2.9. The first part of the matching algorithm, which is data-driven, consists of finding regions in the range image and separately matching them with compatible regions in the object models The second part of the procedure, which is model-driven, sequentially takes single region image-model pairings and searches for more global evidence of the match. That is, adjacent regions are also matched and an overall measure of the adequacy of the match is determined This procedure is equivalent to graph matching, and could prove to be too sensitive to errors in the segmentation of the range map

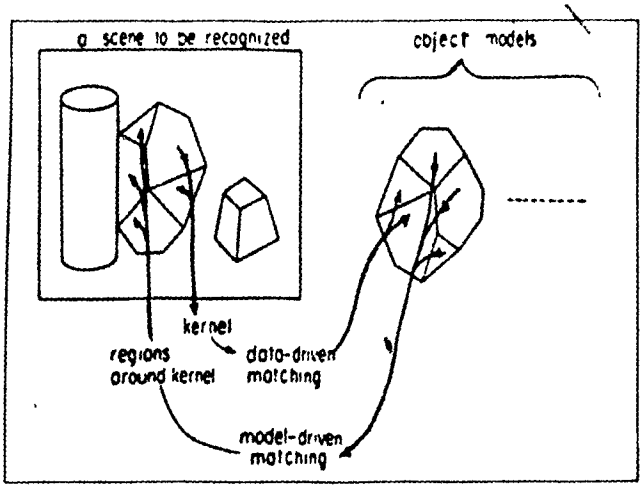


Figure 2.9 Matching process, from [36]

2.2.6 Discussion

At this point, having described a large number of algorithms, it becomes necessary to summarize the work that has been done to date in search of solutions for the

bin-of-parts problem. Thus, a number of remarks can be stated concerning the state-of-the-art in binpicking:

1- Algorithms for the recognition of two-dimensional overlapping objects are reported to work fairly well They use the recognize-pick paradigm, and are usually model-driven.

2- Systems based on sensors of three-dimensional objects, such as range cameras and profile scanners, are most promising since they have the potential for a general solution to the bin-of-parts problem. The systems that have already been built look rather primitive and show considerable room for improvement. However, serious research efforts are now under way, and better systems are therefore to be expected in the near future

3- Holdsite driven algorithms for the recognition of three-dimensional objects using the pick-identify paradigm are very interesting The idea behind this approach is to break the difficult problem of recognizing occluded 3-D objects into two which are easier to handle individually Namely, the problem of finding a suitable holdsite somewhere in the image, and then the problem of identifying an object that has already been isolated from the rest.

The current methods just described also have several drawbacks. Most of the algorithms designed for binpicking deal only with flat objects, and are not easily modifiable for dealing with three-dimensional objects Furthermore, three-dimensional sensors are very expensive, especially when custom made Finally, even the holdsite-based methods previously described have some significant weaknesses:

- Kelley et al [26] use intensity images for holdsite detection, whereas close-in robot control and acquisition are guided by binary optical switches. This method's main disadvantage is that it does not ensure collision-free grasp configurations.

- Boissonnat [7] requires a silhouette in order to produce a list of legal grasps. However, the generation of connected contours is a difficult problem on its own.

- Van Laethem et al [50] use range images to find flat regions which constitute adequate holdsites for a vacuum-type gripper. The constraint that objects must contain flat regions is in our view too restrictive since it excludes a wide range of industrial parts (pipes, bolts, etc)

- Roth and O'Hara [43] employ depth data to find and acquire the highest object in a pile. However, the method by which the orientation of the object's main axis is computed appears to be inadequate, since it involves unnecessary robot motion for range data acquisition purposes

Chapter 3 The BIFOCAL System Description

The BIFOCAL method is designed for binpicking three-dimensional industrial parts using a holdsite-driven approach. Its sensory hardware requirements are only a CCD TV camera and a single-point range finder, both readily available The visual input is used for holdsite detection and fast control of the manipulator, while the depth data provides close-in control

The proposed approach consists of first using computer vision to isolate the location of potential holdsites in the intensity image The robot gripper is then sent to the most promising holdsite using line-of-sight control. Close to the object, the robot is guided by a single-point range finder, and an attempt is made to grasp the unknown object. Upon successful grasping the part is shown to the CCD camera, and its pose is easily computed. Several assumptions are made that simplify ~~the~~ problem, while keeping the solution as general as possible:

(1) The robot hand consists of a parallel-jaw gripper;

(2) The parts to be picked contain at least one region which can be used for grasping by a parallel-jaw gripper (i.e. two parallel clamping surfaces);

(3) The scene is static, meaning that the bin does not move during processing;

(4) The weight of every individual part is within the lifting ability of the robot;

(5) A CCD TV camera is placed over the bin of parts in a fixed position, with its main axis orthogonal to the horizontal plane, and

(6) A single-point range finder is wrist-mounted, so as to provide reliable close-in information about potential holdsites.

In view of these constraints, and in compliance with the selected approach, we designed the BIFOCAL algorithm, which consists of two main steps. The first is concerned with the processing of 2-D images in order to extract parallel lines which constitute potential holdsites in 3-D space. Figure 3 1 shows an example of line detection obtained by applying Mansouri's hypothesis prediction/verification paradigm [34] to the image of a pile of rods. The second step deals with the guidance of the robot to the selected gripping points by using scattered local depth data provided by the range finder. One of the principal objectives of this system is thus to attempt the graceful integration, as visual feedback signals, of the sensory inputs provided by a TV camera and a single-point range finder. These factors result in a system that is considerably more reliable and robust than previous ones.

3.1 The Choice of a Single-Point Range Finder

The four basic material components of our system are: the computer, the robot, the T.V. camera and the range finder, as shown in figure 3.2. For the first three elements

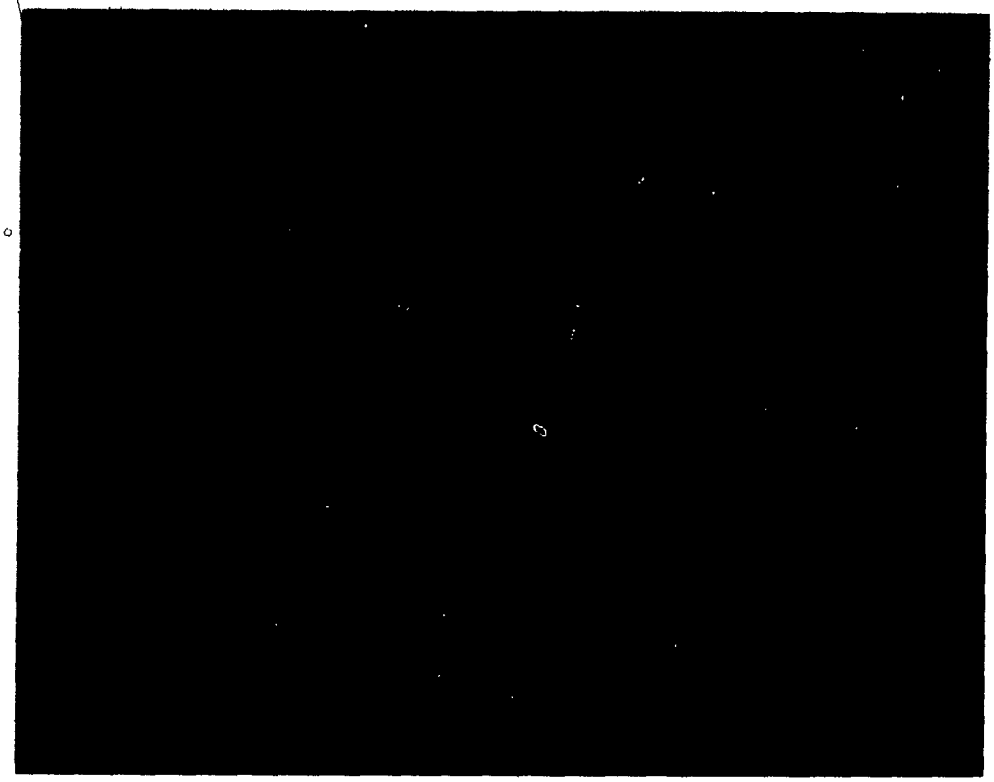


Figure 3.1 Example of line detection

we selected existing equipment at the McRCIM Computer Vision and Robotics Laboratory a Microvax II computer, a Puma 260 robot running RCCL [31], and a Fairchild T V camera However, a single-point range finder had to be selected and acquired This section deals with the problems and issues involved in the choice of a range sensor

The purpose of single-point range finders is to provide accurate measurements of the distance between the measuring device and the object's surface (i e the depth) The specifications that we established for an ideal range finder, as required by our particular application, are the following

- Light weight, since the device is to be wrist-mounted

- Output proportional to depth, analog or digital.

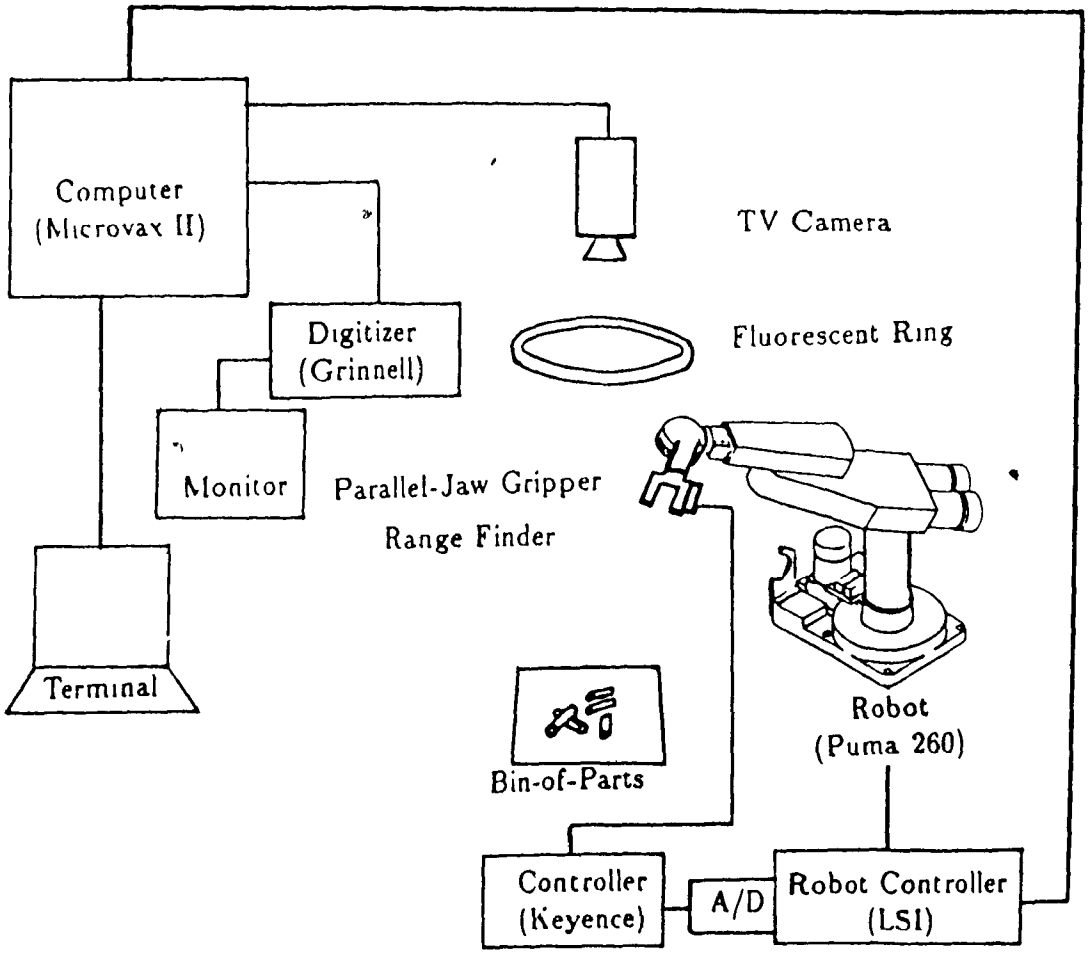


Figure 3.2 The BIFOCAL system

- Large measuring range, from a few mm to some 20 cm.

- Good accuracy, error lower than 0 25 mm

- Small volume ease of installation

- Ease of calibration of the device

The measurements should be independent of variations in color texture, orientation and magnetic properties of the object s surface

Inexpensive

Five different types of range sensors were evaluated photo-electric, triangulation based ultrasonic inductive and capacitive

3 1 1 Photo-Electric Sensors

Photo-electric sensors use modulated infrared light to detect the presence of objects Each sensor contains a light source and a receiver The light source combines an oscillator and a LED, so as to generate modulated light The receiver is comprised of a photo transistor, an amplifier tuned to the frequency of the modulated light, and an output switch Detection occurs when a sufficient amount of light is reflected directly off the object and returned to the receiver, as illustrated in figure 3 3 Thus, the output switch has two possible states ON / OFF

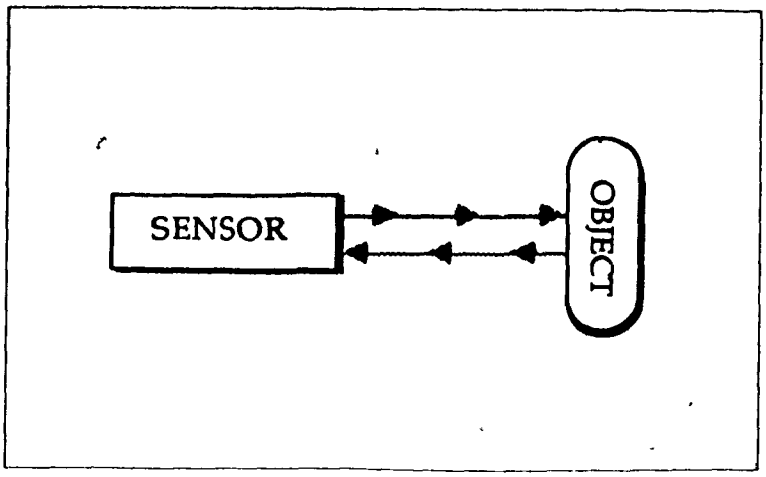


Figure 3.3 Diffuse sensing

Before using this type of sensor a few considerations should be taken into account:

- It only provides a binary output
- Sensing distance depends on the surface reflectivity of the object
- Highly reflective background objects may be detected by the sensor.
- If depth must be measured, then one must use the response curve of the sensor for this surface. A typical curve is shown in figure 3.4

This response curve is ambiguous, since for the same response y there are two possible depth values x_a and x_b (see figure 3.4), which constitutes an important disadvantage.

3.1.2 Triangulation-Based Sensors

Triangulation-based sensors also use infrared light or laser beams. However,

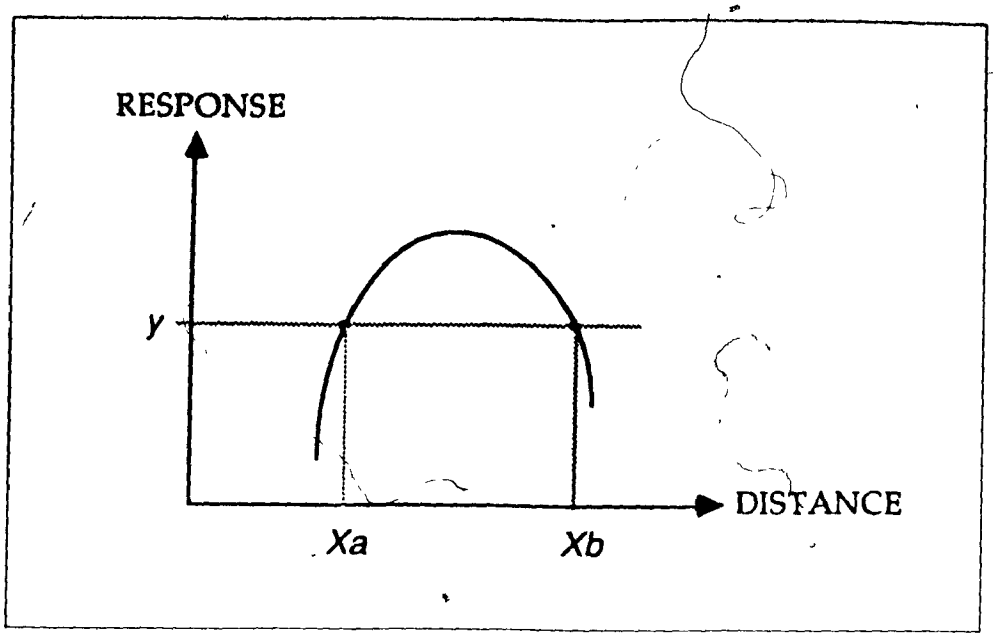


Figure 3.4 Performance chart

they have a different principle of operation, as shown in figure 3.5. They basically consist of an emitter and a linear sensor (a receiver) The emitter projects an IR or laser beam of light, whereas the linear sensor captures the light reflected off the object's surface The depth measurement is a function of where the reflected light hits the sensor.

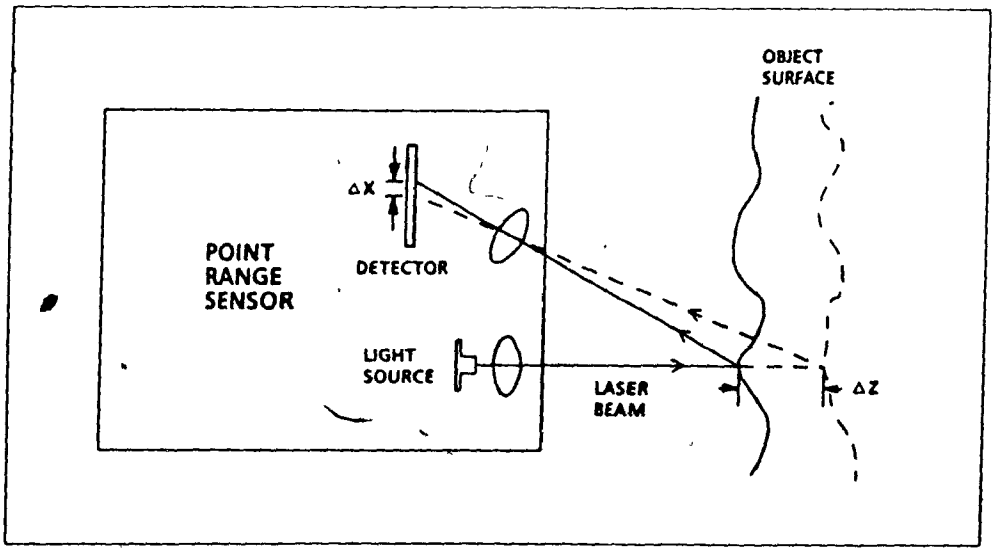


Figure 3.5 The triangulation principle

Triangulation-based sensors are usually very accurate They are also relatively expensive, especially those using laser beams

3.1.3 Ultrasonic Sensors

Ultrasonic sensors use the time-of-flight principle of operation A transducer transmits a short ultrasonic pulse the echo of which is received by the same transducer The time elapsed between the transmission and the reception of the signal is proportional to the distance traveled Since the speed of sound is known the distance between the sensor and the reflecting surface can be calculated

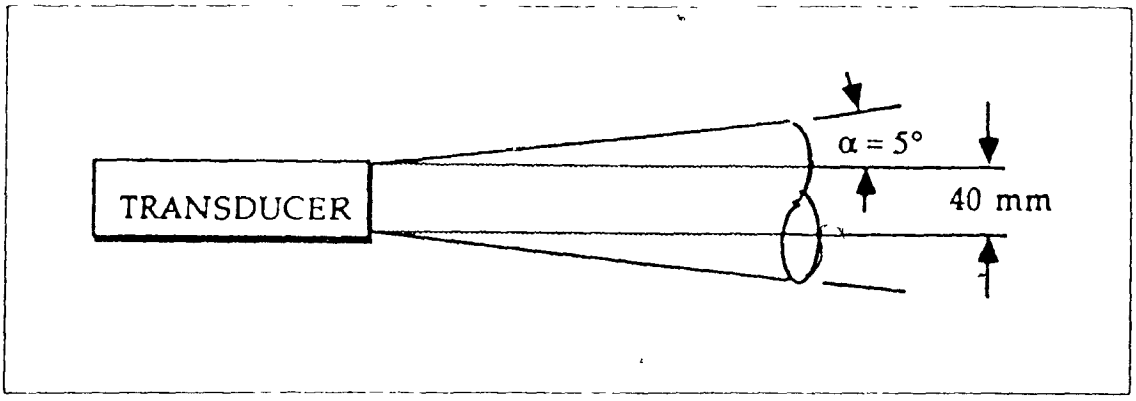


Figure 3 6 Ultrasonic beam pattern

Again a few considerations are worth noting concerning this type of sensor

- They are relatively fragile
- Measuring range goes from approximately 10 *cm* to tens of feet
- Relative accuracy is poor when measuring short distances

- The beam pattern, shown in figure 3.6, is too thick, thus yielding an overly coarse resolution.

3.1.4 Inductive Sensors

Inductive proximity sensors consist of an oscillator and sensing coil, a detector, and an output switch. The oscillator generates an electromagnetic field through the sensing coil When metallic objects enter this field, eddy currents are induced in the objects, causing a voltage drop in the oscillator The detector senses the voltage drop and signals the output to change state

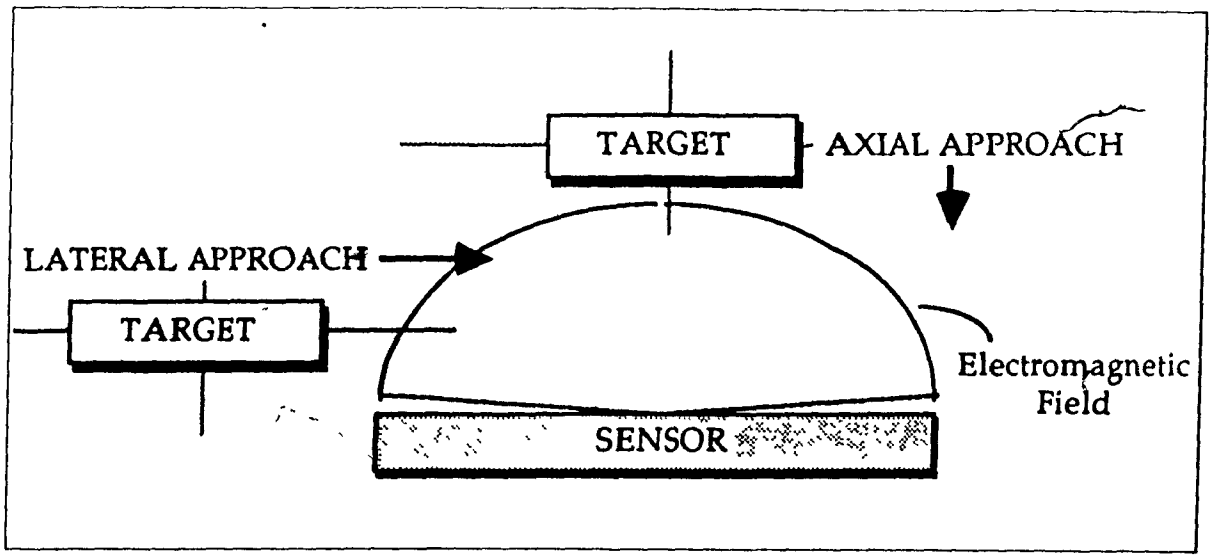


Figure 3.7 Typical sensing field

Figure 3.7 illustrates a typical sensing field for an inductive device. Note that the sensing distance varies for different metals and alloys, and is also a function of object size. Moreover, even if the voltage drop is proportional to the object-sensor distance, there is ambiguity due to the possibility of a lateral, rather than axial, approach.

3.1.5 Capacitive Sensors

Capacitive proximity sensors consist of an oscillator, a capacitor plate, which is the sensing device, and an output switch. The capacitor plate generates an electric field which is altered by the physical (dielectric) properties of the material to be sensed. As an object, composed of elements such as glass, plastic, wood and metal, enters the electric field, capacitance increases bringing about a change in oscillator frequency. The detector senses this frequency variation and outputs a voltage proportional to it. As for the inductive sensors, sensing distance is a function of the sensed material and its size Capacitive sensors are mostly used for binary object detection

3.1.6 Our Choice of a Sensor

After carefully considerating a wide variety of sensors, including Micro Switch's 900 series inductive proximity sensors, Visitronic's HVS electro-optical distance gauges, Candid Logic's Precimeter laser range finder, Skan-a-matic's C40000 series modulated visible beam photo-electric sensors, Tri-tronics' Smarteye infrared photo-electric sensor, Diffracto's Laser Probe 400 laser range sensor, and ISSC's self-contained inductive analog sensor, we selected Keyence's Optical Displacement Sensor *

Keyence's PA-1830 range sensor uses the following principle of operation. an infrared LED beam, narrowed by a lens, is applied to the object Diffused reflection is

---

* Micro Switch, 825 McCaffrey, St-Laurent, Quebec H4T 1N3, Visitronic, P O Box 5077, Englewood, CO 80155, Candid Logic, 31681 Dequindre, P O Box 71943, Madison Heights, MI 48071-0943, Skan-a-matic, Route 5 West, P O. Box S, Elbridge, NY 13060, Tri-tronics, P.O Box 25135, Tampa, Florida 33622, Diffracto, 6360 Hawthorne, Windsor, Ontario N8T 1J9, ISSC, 435 West Philadelphia, P.O. Box 934, York, PA 17405-0934; Keyence, 407 McGill suite 312, Montreal, Quebec H2Y 2G3

focused through a reception lens, forming a spot image on the photo detector. This spot shifts proportionally with object displacement; therefore its position is converted to an electrical signal which is transmitted to the sensor's controller, and subsequently interpreted as a distance.

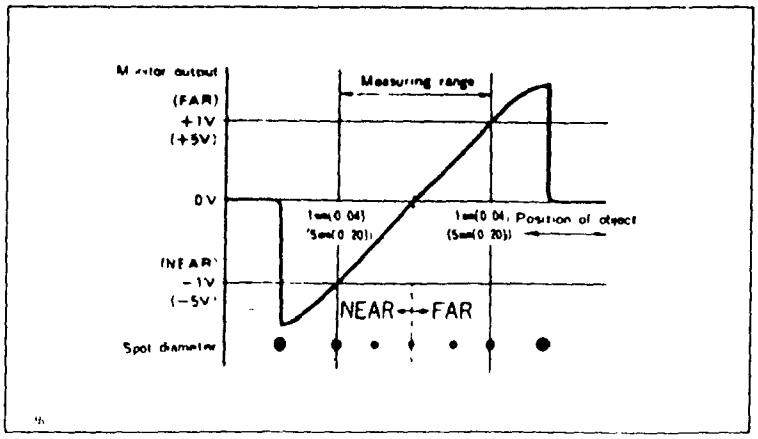


Figure 3.8 Keyence PA-1830 sensor's response curve

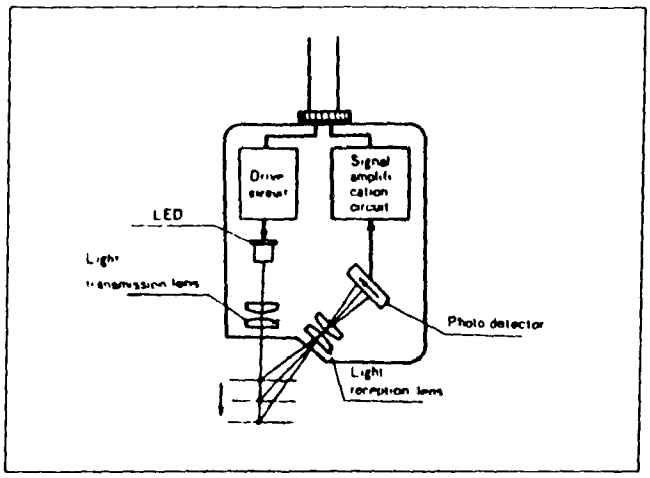


Figure 3.9 Keyence PA-1830 sensor

To summarize, this sensor basically uses an infrared beam and the triangulation principle to provide an output that varies linearly with depth, as shown in figure 3.8. Its specifications are the following:

- Resolution 10 μm
- Analog output, between -5 and $+5$ V
- Output unaffected by material type or object color
- Standoff 40 mm
- Measuring range ± 5 mm
- Accuracy 20 μm + 1% of measurement
- Maximum spot diameter 3 mm
- Weight approximately 80 gr
- Price, approximately 5000 CDN$

The reason for buying this particular sensor is a combination of adequate measuring range, light weight, very good accuracy and reasonable cost

3.2 Lighting

Lighting is a subject too often overlooked by computer vision designers, even though most of them would readily admit that there is no substitute for a high quality image. Pre-processing techniques can be time consuming and do not always yield adequate

results. The selection of appropriate sources of illumination, whenever possible, is thus an important first step in the solution of an image processing problem.

A conflicting consideration is the fact that, when designing a system, one does not want to render it so dependent on illumination that a minor lighting variation would cause the algorithm to fail As a consequence, when the system designer has control over the environment, the best solution is to select lighting equipment so as to obtain the best possible image and, at the same time, allow for illumination variations by designing adaptive, flexible algorithms

In our case, the objective is to detect parallel lines, and we want those lines to correspond to physical edges, not to shadows We therefore selected a fluorescent ring as the sole source of illumination, since it provides with diffuse, almost omnidirectional, lighting Finally, the camera was placed at the center of the fluorescent ring so as to minimize any shadowing effects, as illustrated in figure 3 10

3.3 Hand-Eye Calibration

Hand-eye calibration is necessary for controlling the robot by visual feedback. It consists of a mapping of image coordinates into world (i e robot) coordinates.

Given the fact that axis of the camera is parallel to the vertical axis, we selected a very simple calibration procedure. For any given value of z (i.e. z = constant) image coordinates (u, v) are mapped into world coordinates (x, y), as shown in equations (3.1) and (3.2):

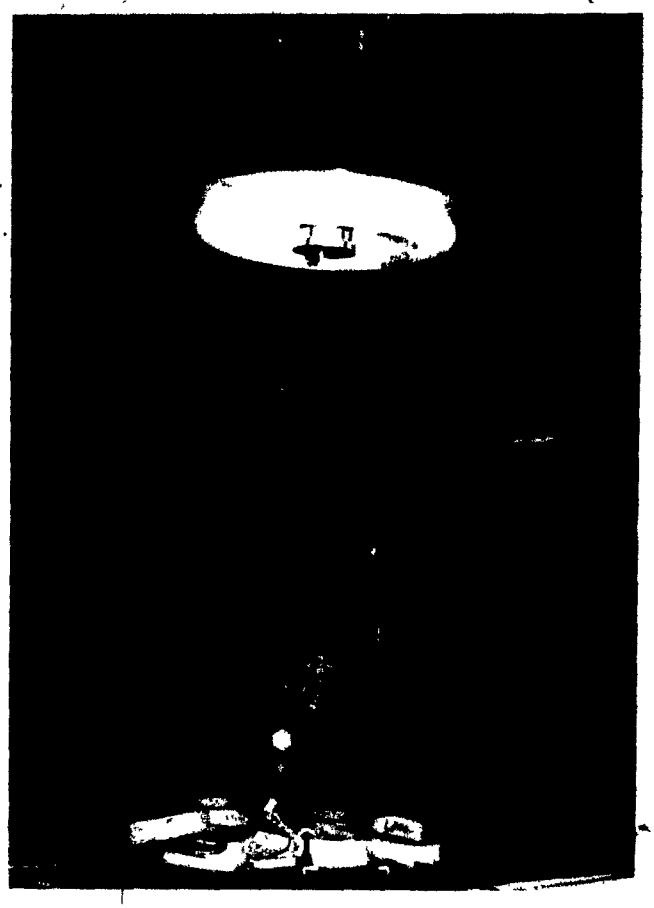


Figure 3.10 Lighting set-up

$$x = \alpha_1 + \alpha_2 u + \alpha_3 v \tag{3.1}$$

$$y = \beta_1 + \beta_2 u + \beta_3 v \tag{3.2}$$

where $\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2$ and β_3 are calibration coefficients. The image-world mapping therefore consists of a translation and a rotation. Obviously these coefficients are only valid for a specific z = constant plane, which is referred to as the calibration plane. Also to be computed are the heights of the calibration plane and of the camera, since they allow the interpolation of the calibration results over the entire volume defined by the camera's

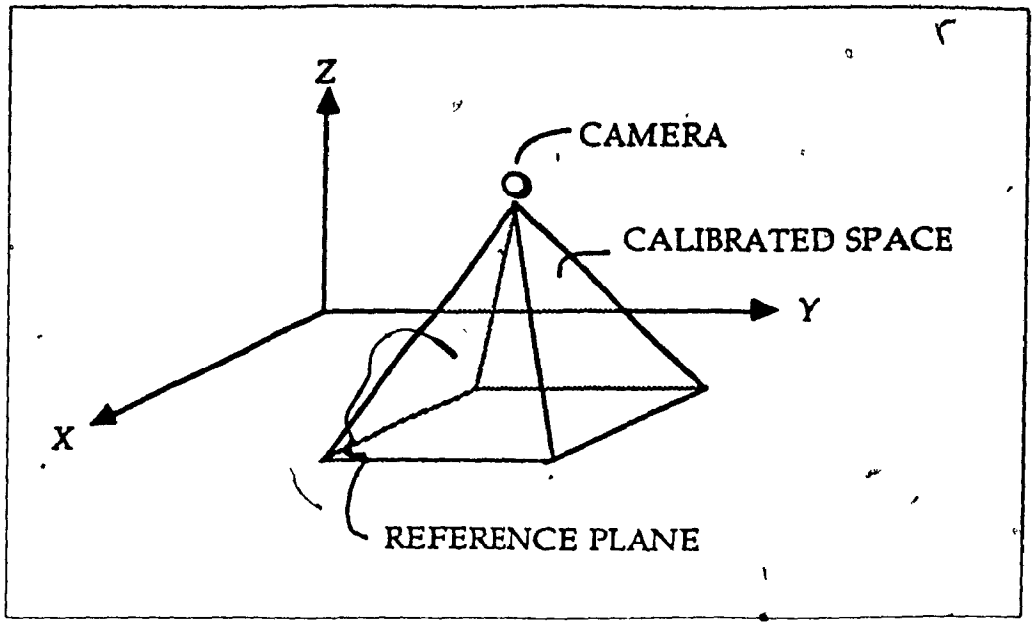


Figure 3.11 Calibrated space

field of view, as illustrated in figure 3.11. Hence, eight calibration parameters are required to map any image point (u,v) into its corresponding line in x-y-z space

Since only eight unknowns have to be determined, four points would be sufficient to obtain a single solution. By point we refer to the camera coordinates (u,v) and the corresponding world coordinates (x,y,z). However, calibration accuracy can be considerably improved by collecting more points so as to generate an overconstrained system of equations, and then solve it using a least-squares fit. We employed this latter technique with about 20 points. The latter are collected by first connecting to the robot's end effector a special calibration tool, shown in figure 3.12, that contains regularly spaced marks on its surface, and subsequently showing it to the TV camera. The operator then manually locates those marks in the image, and each one of them is saved as a (u,v,x,y,z) vector. Once the desired number of points has been collected, the calibration parameters are computed by solving the underlying system of equations. The results obtained indicate that the average error is approximately 0.33 mm with a standard deviation of 0.16 mm.

These values are well within the tolerances that can be accounted for by the use of the range sensor, which confirms the increased flexibility and robustness brought about by this additional source of sensory input.

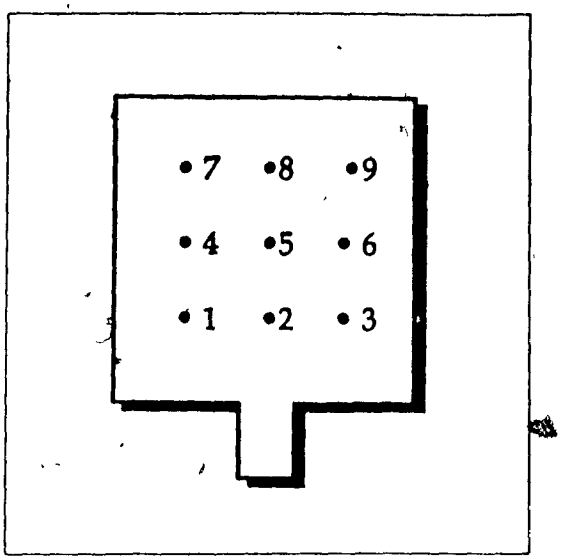


Figure 3.12 Calibration tool

Chapter 4 Holdsite Determination Using a TV Camera

This chapter describes how holdsites are found in the intensity image There are two main steps in finding potential holdsites in a two-dimensional image The first consists of a line detection procedure, whereas the second finds holdsites and measures their quality and appropriateness

4.1 Line Detection

The line detection algorithm has been developed at McGill University by Mansouri [34] and uses a hypothesis prediction / verification paradigm Given a pixel (x_c, y_c) in the image, whose gradient, as computed using the Sobel operator, exceeds a pre-determined threshold, it is hypothesized that a segment of a line exists which is centered at the (x_c, y_c) pixel, whose orientation is perpendicular to that of the gradient, and whose total length is equal to $2n+1$ points (where n is the order of the segment) Thus, a set of points $\{(x_i, y_i)\}$ is assumed to belong to the hypothesized line, as shown in equation (4 1)

$$H : \forall (x_i, y_i) \in S \; . \; (x_i - x_c)e_{x_c} + (y_i - y_c)e_{y_c} = 0 \qquad (4\,1)$$

where H is the hypothesis, S the segment, and (e_{x_c}, e_{y_c}) the gradient components along the x and y directions at pixel (x_c, y_c)

The sample mean orientation of the gradient through the segment are given by $\overline{S_x}$ and $\overline{S_y}$ (see equations (4 2) and (4 3))

$$\overline{S_x} = \frac{1}{2n+1} \sum_{i=1}^{2n+1} e_{x_i} \tag{4 2}$$

$$\overline{S_y} = \frac{1}{2n+1} \sum_{i=1}^{2n+1} e_{y_i} \tag{4 3}$$

In a similar manner sample variances S_x^2 and S_y^2 are given by equations (4 4) and (4 5)

$$S_x^2 = \frac{1}{2n} \sum_{i=1}^{2n+1} (e_{x_i} - \overline{S_x})^2 \tag{4 4}$$

$$S_y^2 = \frac{1}{2n} \sum_{i=1}^{2n+1} (e_{y_i} - \overline{S_y})^2 \tag{4.5}$$

If we assume that $\{e_{x_i}\}$ and $\{e_{y_i}\}$ are normally distributed random variables with unkown mean and variance, namely $\{\mu_x, \mu_y\}$ and $\{\sigma_x^2, \sigma_y^2\}$, the sampling distributions of the statistics given by equations (4.6) and (4.7) are then student t distributions with $2n$ degrees of freedom.

$$\frac{\overline{S_x} - \mu_x}{\sqrt{\frac{S_x^2}{2n+1}}} \tag{4.6}$$

$$\frac{\overline{S_y} - \mu_y}{\sqrt{\frac{S_y^2}{2n+1}}} \tag{4.7}$$

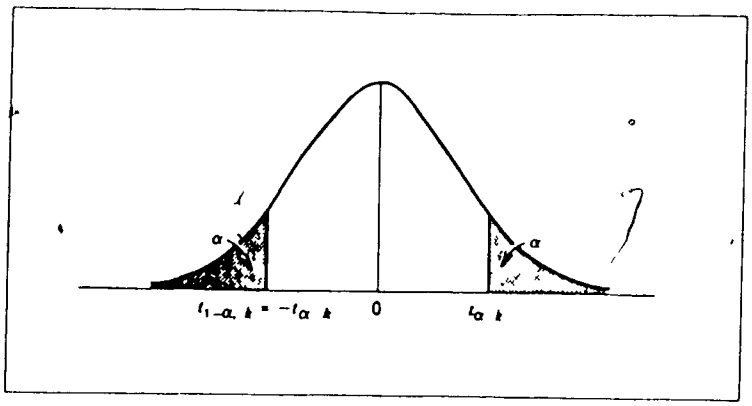


Figure 4.1 Student t distribution

Therefore, it is possible to show that a $100(1-\alpha)$ percent two-sided confidence interval on μ_x is given by equation (4.8):

$$\overline{S_x} - t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_x^2}{2n+1}} \leq \mu_x \leq \overline{S_x} + t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_x^2}{2n+1}} \tag{4.8}$$

This also holds for the confidence interval on μ_y (see equation (4.9)):

$$\overline{S_y} - t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_y^2}{2n+1}} \leq \mu_y \leq \overline{S_y} + t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_y^2}{2n+1}} \tag{4.9}$$

Maintaining the same assumptions, it is also possible to prove that the sampling distributions of the statistics given by equations (4.10) and (4.11) are chi-squared distributions with $2n$ degrees of freedom.

$$\frac{2nS_x^2}{\sigma_x^2} \tag{4.10}$$

$$\frac{2nS_y^2}{\sigma_y^2} \tag{4.11}$$

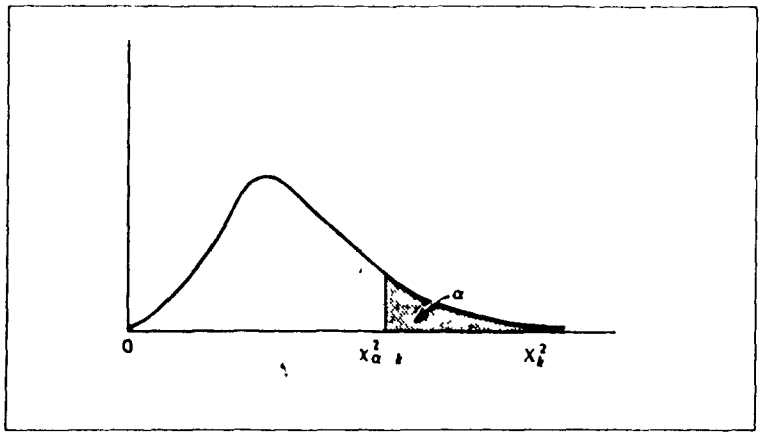


Figure 4.2 Chi-Square distribution

Hence, we can show that a $100(1-\alpha)$ percent two-sided confidence interval on σ_x^2 is given by equation (4.12):

$$\frac{2nS_x^2}{X^2_{\frac{\alpha}{2},2n}} \leq \sigma_x^2 \leq \frac{2nS_x^2}{X^2_{1-\frac{\alpha}{2},2n}} \tag{4.12}$$

Again, the same analysis is also true for σ_y^2, as shown in equation (4.13):

$$\frac{2nS_y^2}{X^2_{\frac{\alpha}{2},2n}} \leq \sigma_y^2 \leq \frac{2nS_y^2}{X^2_{1-\frac{\alpha}{2},2n}} \tag{4.13}$$

Thus, in order to establish bounds for the variance and mean difference for $100(1-\alpha)\%$ confidence intervals on the estimation of parameters, namely μ_x, μ_y, σ_x^2 and σ_y^2, we use the results stated above. This procedure yields, for μ_x, a bound on the error D_x, which is computed as shown in equation (4.14)

$$D_x = max\left(\left|e_{x_c} - \overline{S_x} - t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_x^2}{2n+1}}\right|, \left|e_{x_c} - \overline{S_x} + t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_x^2}{2n+1}}\right|\right) \tag{4.14}$$

Similarly, D_y is computed as shown in equation (4.15)

$$D_y = max\left(\left|e_{y_c} - \overline{S_y} - t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_y^2}{2n+1}}\right|, \left|e_{y_c} - \overline{S_y} + t_{\frac{\alpha}{2},2n}\sqrt{\frac{S_y^2}{2n+1}}\right|\right) \tag{4.15}$$

Bounds on the variance, V_x and V_y, are also computed in a simple way, as illustrated in equations (4.16) and (4.17)

$$V_x = \frac{2nS_x^2}{X^2_{1-\frac{\alpha}{2},2n}} \tag{4.16}$$

$$V_y = \frac{2nS_y^2}{X^2_{1-\frac{\alpha}{2},2n}} \tag{4.17}$$

At this point, we can state four tests of hypotheses for the purpose of confirming or rejecting the existence of one specific line:

$$H_0\ \mu_x = \mu_{x_0} \quad \textit{as opposed to} \quad H_4\ \mu_x \neq \mu_{x_0} \qquad (4.18)$$

σ_x^2 *unknown*

$$H_1\ \mu_y = \mu_{y_0} \quad \textit{as opposed to} \quad H_5\ \mu_y \neq \mu_{y_0} \qquad (4.19)$$

σ_y^2 *unknown*

$$H_2\ \sigma_x^2 = \sigma_{x_0}^2 \quad \textit{as opposed to} \quad H_6\ \sigma_x^2 \neq \sigma_{x_0}^2 \qquad (4.20)$$

$$H_3\ \sigma_y^2 = \sigma_{y_0}^2 \quad \textit{as opposed to} \quad H_7 : \sigma_y^2 \neq \sigma_{y_0}^2 \qquad (4.21)$$

Hypotheses H_0 through H_3 have to be true in order to infer the presence of the line. Therefore, line L exists if and only if the following expression is true:

$$(V_x \leq \Omega_x^2)\ \textit{and}\ (V_y \leq \Omega_y^2)\ \textit{and}\ (D_x \leq \Delta_{\mu_x})\ \textit{and}\ (D_y \leq \Delta_{\mu_y}) \qquad (4.22)$$

where Ω_x^2 and Ω_y^2 are user-determined thresholds on variance, and Δ_{μ_x} and Δ_{μ_y} are the thresholds on mean differences, which are also selected by the user.

This line detection algorithm finds lines in a sequential manner. Thus in order to diminish the number of lines generated by the same physical edge, which is in part due to the fact that edge operators tend to thicken edges, a new line is not created if it is located in the neighborhood of previously found lines This is stated in a more rigorous way by equation (4 23)

$$S_j \quad S_i < \Rightarrow Card(S_j \bigcap N(S_i)) \quad \lambda Card(S_j) \tag{4.23}$$

where S_j is the new line and S_i a previously found segment N is the neighborhood defined as a 3-pixel wide line whose center and orientation coincide with those of the line for which it is determined λ is a user selected coefficient that sets the threshold with regard to the maximum amount of overlap between new and previous lines so that the new line can be saved as such

When lines are found in the intensity image they are stored in order on the basis of their orientation This contributes to the subsequent processing of the lines which analyses certain relationships between them so as to detect more elaborate geometrical structures

4.2 Holdsite Finding

Once the lines have been found and stored according to their orientation it becomes necessary to detect the potential holdsites that they generate. A priori, any combination of two lines in the database may constitute a holdsite These large number of possible combinations must therefore be pruned so as to retain only those pairs of lines that

are most likely to correspond to potential holdsites. Hence, the holdsite finding procedure consists of a search algorithm.

Three parameters applied to pairs of lines are used for pruning the search space: orientation, distance and separation Since the algorithm searches for parallel clamping surfaces, the orientation constraint can be stated as in equation (4.24), and illustrated as in figure 4 3. This constraint is used to reject line pairs whose orientations are not opposite (i e. approximately 180^o apart)

$$(\alpha_1 + 175^o)_{360} < \alpha_2 < (\alpha_1 + 185^o)_{360} \tag{4.24}$$

where α_1 is the orientation angle of the first line L_1 , α_2 is the angle of the second line L_2 , and the values are modulo 360

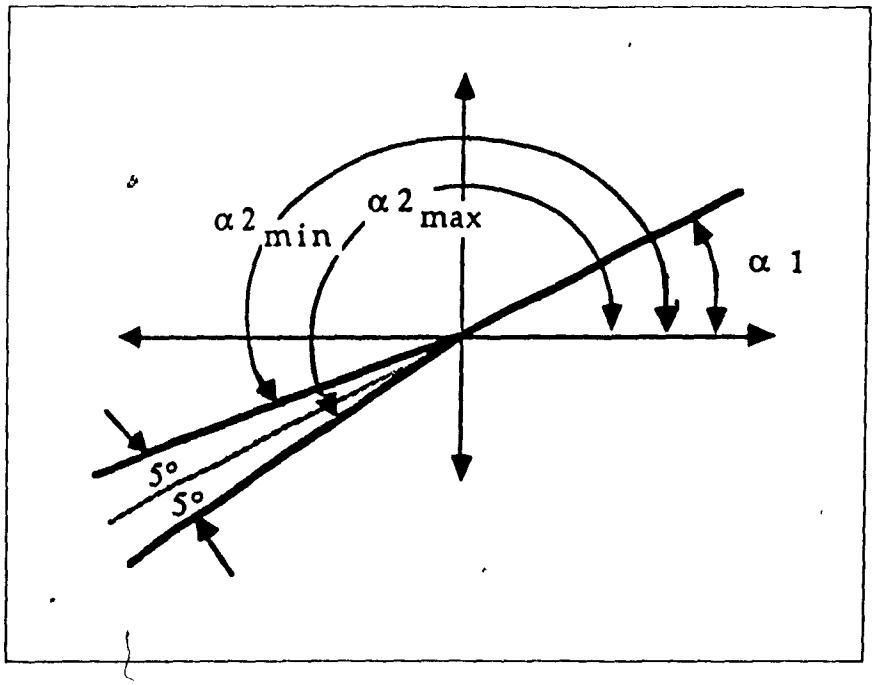


Figure 4.3 The orientation constraint

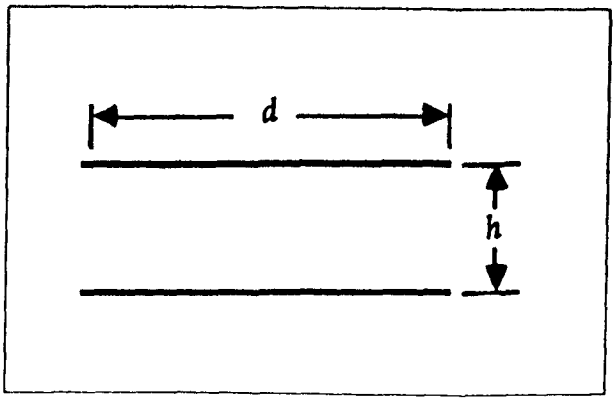


Figure 4.4 Geometric model of the holdsite

Now, considering the geometric model of the holdsite shown in figure 4.4, we can establish the remaining two constraints. The first relates to the distance between the centers of the two line segments L_1 and L_2. This distance constraint, as illustrated in figure 4.5, causes the pruning of those pairs composed of lines which are too far apart. Specifically, potential holdsites are rejected if the distance between the centers of gravity of the two lines exceeds a computed threshold, as indicated in equation (4.25).

$$d(L_1, L_2) \leq D_{max} \tag{4.25}$$

where $d(L_1, L_2)$ is the distance between lines, D_{max} is the maximum allowed distance (see equation (4.26)), and d and h are the dimensions of the holdsite.

$$D_{max} = \sqrt{\frac{d^2}{4} + h^2} \tag{4.26}$$

The last is the separation constraint. It is based on the average separation between the two segments, as defined in figure 4.6. This separation must be in the range indicated by equation (4.27):

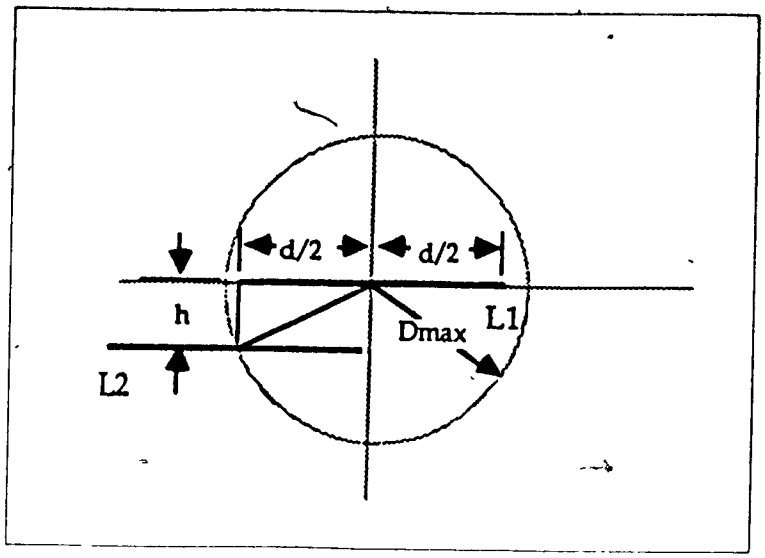


Figure 4.5 The distance constraint

$$0.75h < s < 1.25h \tag{4.27}$$

where s is the average separation, and h the holdsite width

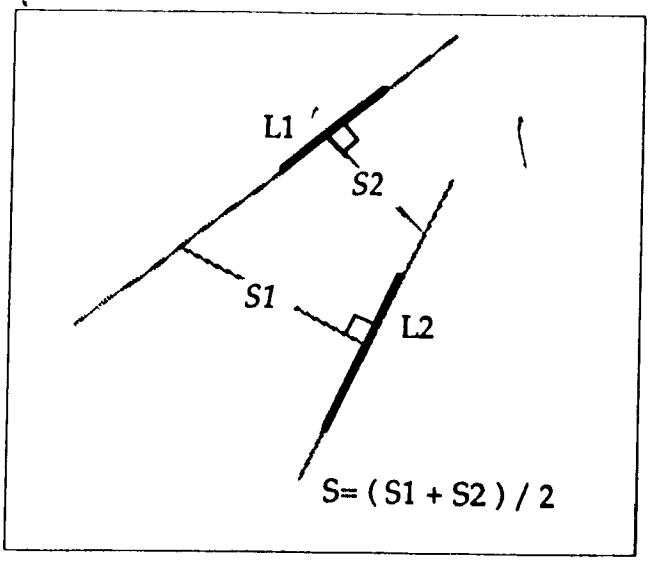


Figure 4.6 The separation constraint

The separation constraint therefore consists of matching procedure between the line pairs

and the holdsite model, since it allows the algorithm to retain only those pairs of lines which match, at least to a certain degree, the underlying model.

The constraints described above permit the search algorithm to build a list of potential holdsites which, as in the case of lines, are ordered according to their orientation. These holdsites will be evaluated according to suitability criteria which we will describe later However, we must first deal with the problem of multiple representations for the holdsites

4.3 Holdsite Filtering

The algorithm hypothesizes potential holdsites which may not be unique for each actual holdsite In other words, the same physical holdsite may correspond to many computed holdsites For reasons of efficiency and logical consistency, we must therefore filter the redundant potential holdsites so that there is a one-to-one mapping between the physical world and a representation of it In view of this, a filtering procedure has been implemented in which holdsites are clustered on the basis of location and orientation. Two holdsites are merged together if the distance between their centers of gravity is smaller than a model-based threshold (see equation (4 28)) and if their respective orientations are approximately equal, as stated in equation (4 29)

$$(cg_{x_1} - cg_{x_2})^2 + (cg_{y_1} - cg_{y_2})^2 < \left(\frac{d}{2}\right)^2 \tag{4.28}$$

$$(\beta_1 - 2.5^o) < \beta_2 < (\beta_1 + 2.5^o) \tag{4.29}$$

where (cg_{x_1}, cg_{y_1}) and (cg_{x_2}, cg_{y_2}) are the centers of gravity of the first and second holdsites, respectively, and β_1 and β_2 are their corresponding orientations (see figure 4.7)

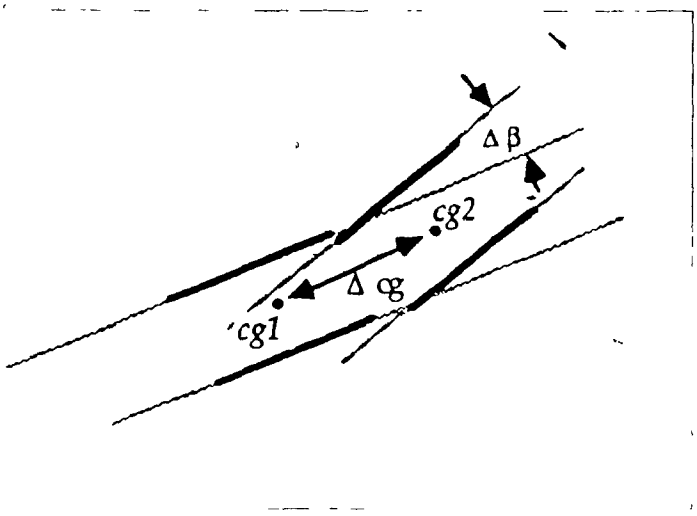


Figure 4.7 Holdsite merging criteria

After filtering, the remaining holdsites are all distinct (i.e. they correspond to different local image structures). At this point, these well-defined potential gripping points must be evaluated in terms of their quality and appropriateness. But before explaining the manner in which quality is computed, we describe the characteristics of a holdsite in terms of what is desirable, of what makes a holdsite a good one, and of the risks involved in grasping an object with a parallel-jaw gripper. It then becomes clear which parameters should be used as a measure of quality.

4.4 Characteristics of a Holdsite

In this section we define four basic properties of a holdsite that characterize its suitability to a parallel-jaw gripper, namely slippage, stability, accessibility and safety.

4.4.1 Slippage

Slippage measures the probability that the object may slip out of the robot hand during acquisition [43]. In the intensity image, the only clue about this parameter is given by the alignment of the two clamping surfaces, which indicates the possibility of slippage in the x-y plane as the gripper attempts grasping (see figure 4.8). Thus we have:

$$Slippage \propto \theta \tag{4.30}$$

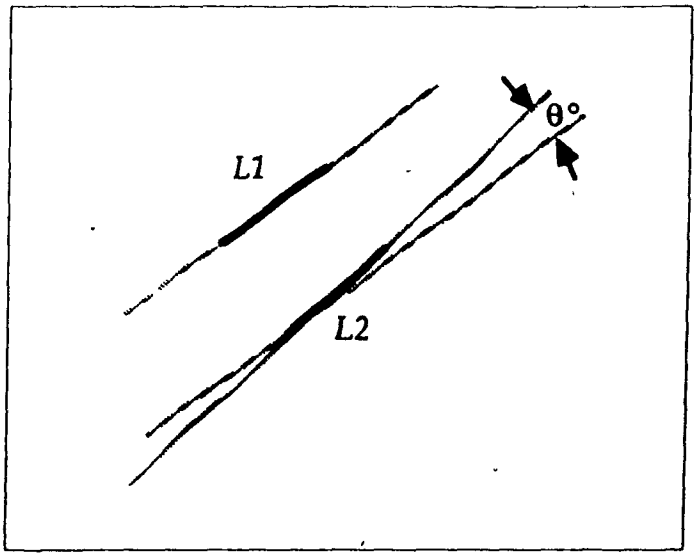


Figure 4.8 Slippage in the 2-D image

4.4.2 Stability

Stability during part acquisition is directly related to the size of the contact area between the part and the manipulator's fingers. In the intensity image, stability can only be approximated by assuming that it is proportional to the overlap of the projection of the

first line onto the second line that constitutes the potential holdsite, as illustrated in figure 4.9. Thus we can state that:

$$Stability \propto overlap \tag{4.31}$$

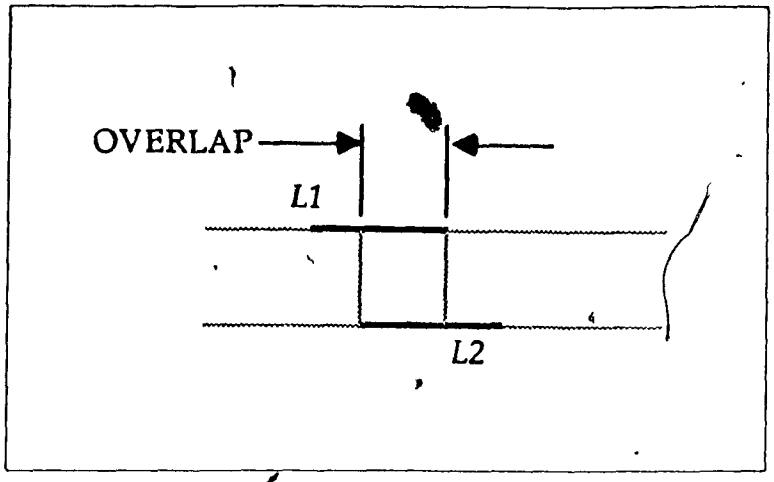


Figure 4.9 Stability in the 2-D image

4.4.3 Accessibility

For a specific holdsite and a given gripper, accessibility can be characterized as the binary decision as to whether the holdsite can be reached or not. A holdsite may be located in an intensity image, but it might be impossible to grasp it, as neighboring objects might impede acquisition. The height of these potential obstacles cannot be determined in an intensity image and thus neither can holdsite accessibility.

4.4.4 Safety

Safety is determined by the effects that uncertainties in the translational and

rotational location of the gripper may have on the acquisition process, such as the possibility of collisions or grasping failures

Once a suitable holdsite has been detected, the grasping pose of the gripper is computed It consists of six values, the 3-D position (x, y, z) and the orientation angles $(roll, pitch, yaw)$ Safety is the sensitivity of the stability and accessibility of the holdsite to small changes in x y z $roll$, $pitch$ and yaw Thus

$$Safety \propto \frac{\delta Stability}{\delta x} \tag{4 32}$$

$$Safety \propto \frac{\delta Accessibility}{\delta x} \tag{4 33}$$

If we reduce this to the x y plane, as is the case in intensity images, safety is defined by the amount of displacement that the grasp configuration can tolerate along the x and y directions and about the z axis

4.5 Computation of Holdsite Quality

In light of the matters discussed in the previous section, holdsite quality depends on the closeness of fit between the data and the holdsite model, and on the holdsite characteristics, namely slippage stability, accessibility and safety

As shown previously, holdsite accessibility cannot be computed on the basis of a two-dimensional image Furthermore, safety can be accounted for by incorporating it into the slippage and safety computations This is achieved by adding a safety factor that

results in more conservative estimates. Quality is therefore evaluated as a function of three parameters: closeness of model fit c_f , slippage s_l and stability s_t , as given in equation (4.34):

$$q = 100 - \alpha c_f - \beta s_l - \gamma s_t \tag{4.34}$$

where q indicates the quality in percentage, α, β and γ are constant parameters, c_f is the difference between nominal and computed holdsite separation, s_l is the difference between the orientations of the two segments that constitute the holdsite, and s_t is the shift angle as defined in figure 4.10

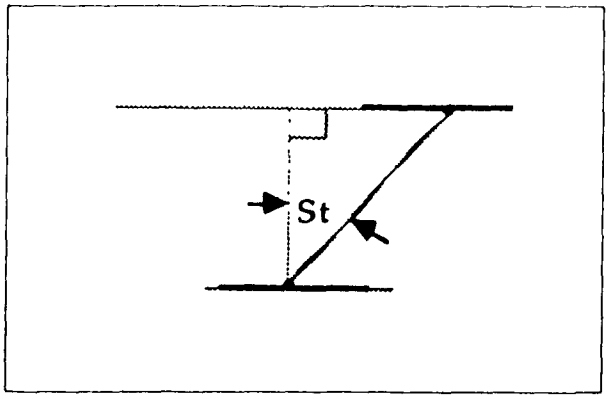


Figure 4.10 The shift angle

It is worth noting that c_f equals zero for a perfect fit between model and data, s_l is also equal to zero if both holdsite lines have the same orientation, and finally s_t has a value of zero if the two lines are exactly facing each other, thereby indicating excellent stability. Therefore a quality of 100 % is assigned to a perfect holdsite and this value decreases as the holdsite characteristics become less desirable.

The program computes the quality q of every filtered holdsite and selects the

three best potential holdsites whose quality is above a threshold (20 %). These ordered gripping points constitute the input to the line-of-sight module which controls the robot's approach and the subsequent range scanning procedure. The line-of-sight approach consists of sending the robot's gripper towards the selected holdsite following a specific line in space. This line (i e. the line-of-sight) is defined by the focal point of the camera and by the center of gravity of the holdsite, as illustrated in figure 4 11.

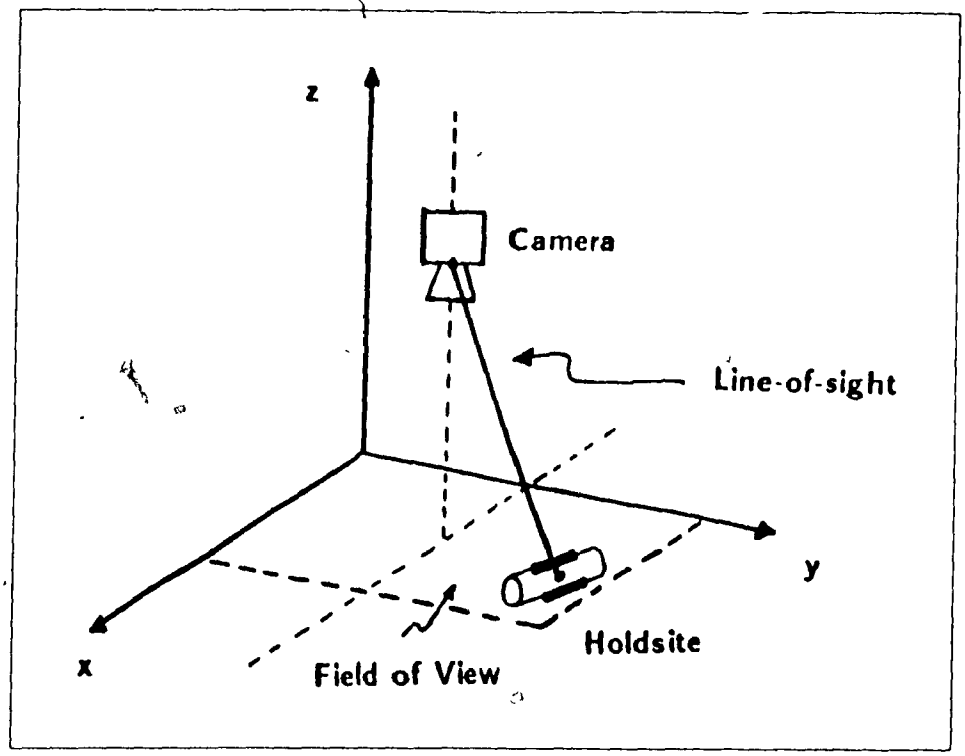


Figure 4.11 The line-of-sight

The robot, which was positioned outside the camera's field of view during image acquisition and processing, is moved to the line-of-sight at a pre-determined height. The gripper orientation is selected so as to result in the range finder being aligned with the line-of-sight and pointing towards the holdsite center, as shown in figure 4 12. As for the orientation about the line-of-sight, it is determined in order to position the gripper fingers parallel to the clamping surfaces of the holdsite.

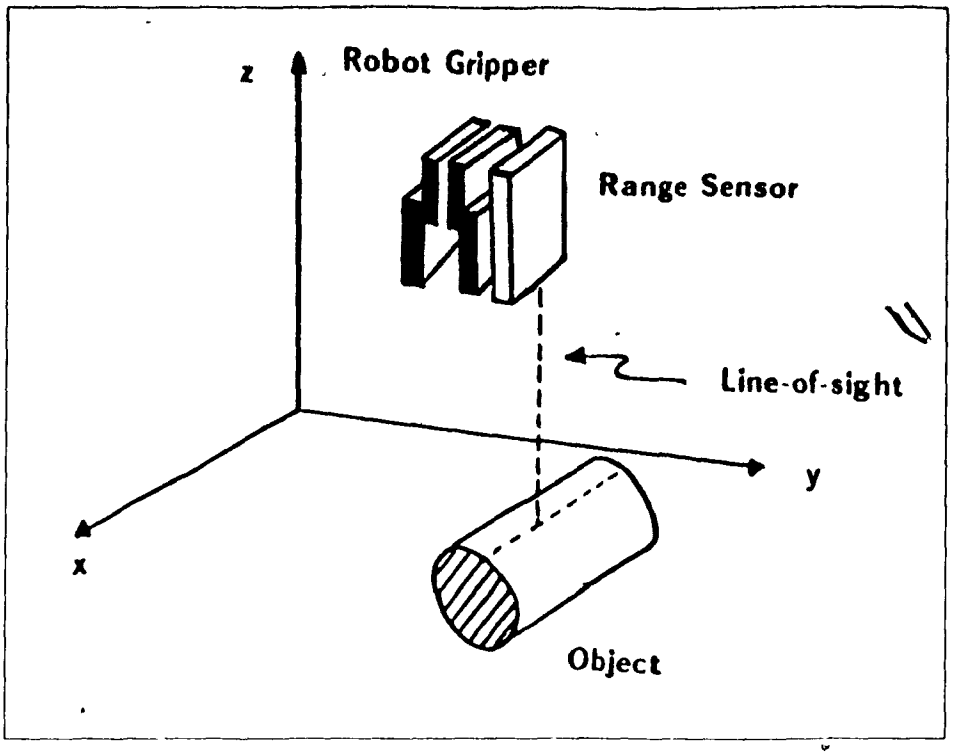


Figure 4.12 Gripper orientation during line-of-sight approach

At this point, the second step of the BIFOCAL VISION algorithm must be carried out This step encompasses the guarded approach and the acquisition and processing of local range data

Chapter 5 Range Processing

This chapter describes the guarded approach and the range image processing techniques used in order to confirm holdsite existence, update the location of its center of gravity and orientation, and determine whether the holdsite can be reached and grasped by the robot gripper

5.1 The Guarded Approach

The objective of the guarded approach is to move the robot hand towards the holdsite while at the same time preventing any collisions from occurring This is achieved by collecting range data as the robot approaches the holdsite and stopping the motion on condition (i e when the wrist-mounted sensor is at a certain distance from the object)

The guarded approach is implemented in two steps due to practical considerations concerning the range sensor, as indicated in section 3 1 6 and illustrated in figure 3 8 During the first phase the robot moves at a high speed along the line-of-sight until the first positive indication is returned by the sensor. This takes place at approximately 50 mm from the holdsite. In the second phase the robot continues moving in the same

direction but at a much lower speed. This fine approach ends when the sensor is at exactly 35 mm from the object, thus placing the holdsite within the sensor's measuring range so that a local depth grid can be acquired by moving the sensor at constant height. In other words, the guarded approach determines the height at which the robot end effector should stay throughout the whole range data acquisition process. This has the effect of ensuring that the holdsite is always within the range finder's measuring range.

The range grid consists of four parallel profiles and has a rectangular shape The location of the grid on the $x - y$ plane is chosen as a function of the holdsite location computed using the global intensity image, more precisely the center of the range grid is selected so as to coincide with the center of gravity of the 2-D holdsite The grid's orientation is chosen so that the collected profiles are perpendicular to the holdsite, as illustrated in figure 5 1 The actual size of the grid is a function of the holdsite width h and gripper thickness. Another factor is the required tolerance with regard to possible holdsite location errors due to the 2-D image analysis In our case we chose to use a 12.5 mm wide by 25.0 mm long grid, the length being measured along the profile These dimensions ensure, for our specific parts and gripper, proper holdsite detection and accessibility computation.

5.2 Range Image Processing

Once the data have been acquired by moving the robot end effector in a specific way so as to generate a local depth grid (see figure 5.1), it becomes necessary to process this range grid in order to confirm the presence of the potential holdsite, update its location, and compute its *accessibility* by a parallel-jaw gripper. This procedure is summarized in the diagram shown in figure 5.2.

Figure 5.3 shows such an accessible holdsite. It is defined by two clamping surfaces separated by a specific distance as determined by the holdsite model, having specific areas of free space in front of each of the two surfaces. The size a_u of these areas is related to the size of the robot fingers. In our case we selected a_u equal to three times the finger width (i e $a_u = 5$ mm) as a safety precaution.

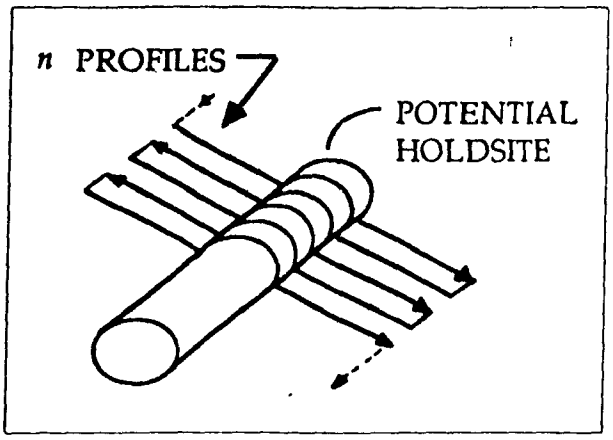


Figure 5.1 Path followed by the range sensor's beam as the robot end effector moves so that the sensor can collect a grid of depth values

As indicated above the range profiles are collected perpendicular to the main axis of the holdsite. This can be done since the *approximate* holdsite location and orientation are previously found in the intensity image. The range grid can therefore be processed row by row. Every row corresponds to a different profile of depth values and is processed as a one-dimensional digital signal so as to extract the location of the two clamping surfaces that constitute the holdsite. The profile processing procedure is illustrated, through two examples, in figure 5.4.

The profiles are combined, because of their spatial contiguity, into an integrated description of the grid. Profile processing is performed according to the following steps:

- *Smoothing:* Implemented by a Gaussian mask [30] which is applied over a neighbor-

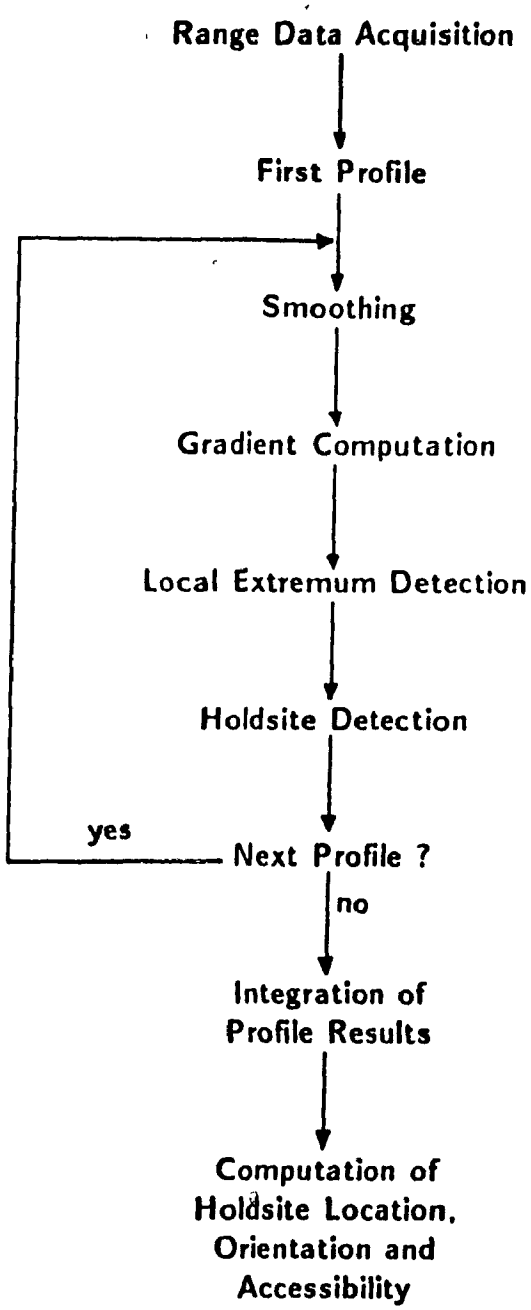


Figure 5.2 Diagram of the range image processing procedure

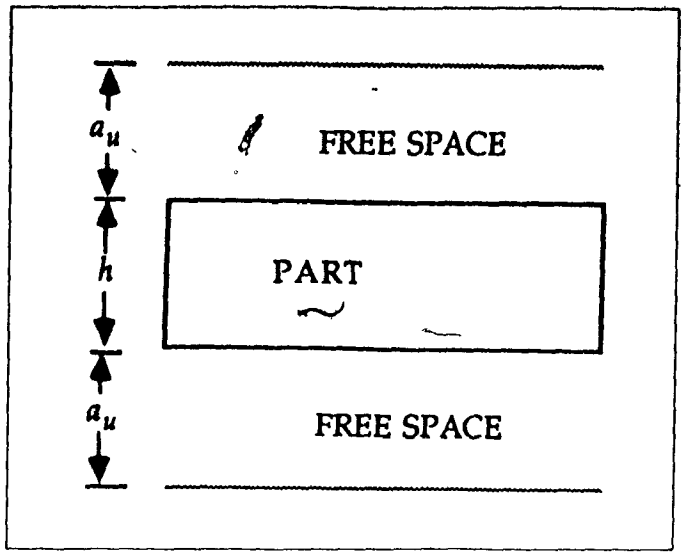


Figure 5.3 An accessible holdsite

hood of five pixels. Its purpose is to eliminate small artifacts and other types of noise generated during the range data acquisition process.

- *Gradient computation*: Implemented as a subtraction of the values of adjacent pixels along a given row of the smoothed image.

- *Local extremum detection:* The objective is to detect the presence and location of potential clamping surfaces which are assumed to correspond to extrema in the signal's first derivative (see figure 5.4).

- *Holdsite detection:* This step consists of detecting the first maximum-minimum pair that matches the holdsite width, as discussed in Section 4.2.

When all of the profiles have been processed, we obtain a set $\{z_k(j_m), z_k(j_p)\}$, where $k = 1, 2, ..., n$, n is the number of profiles, z is the depth, j_m is the location of the local maximum, and j_p is the location of the matching local minimum. In other words,

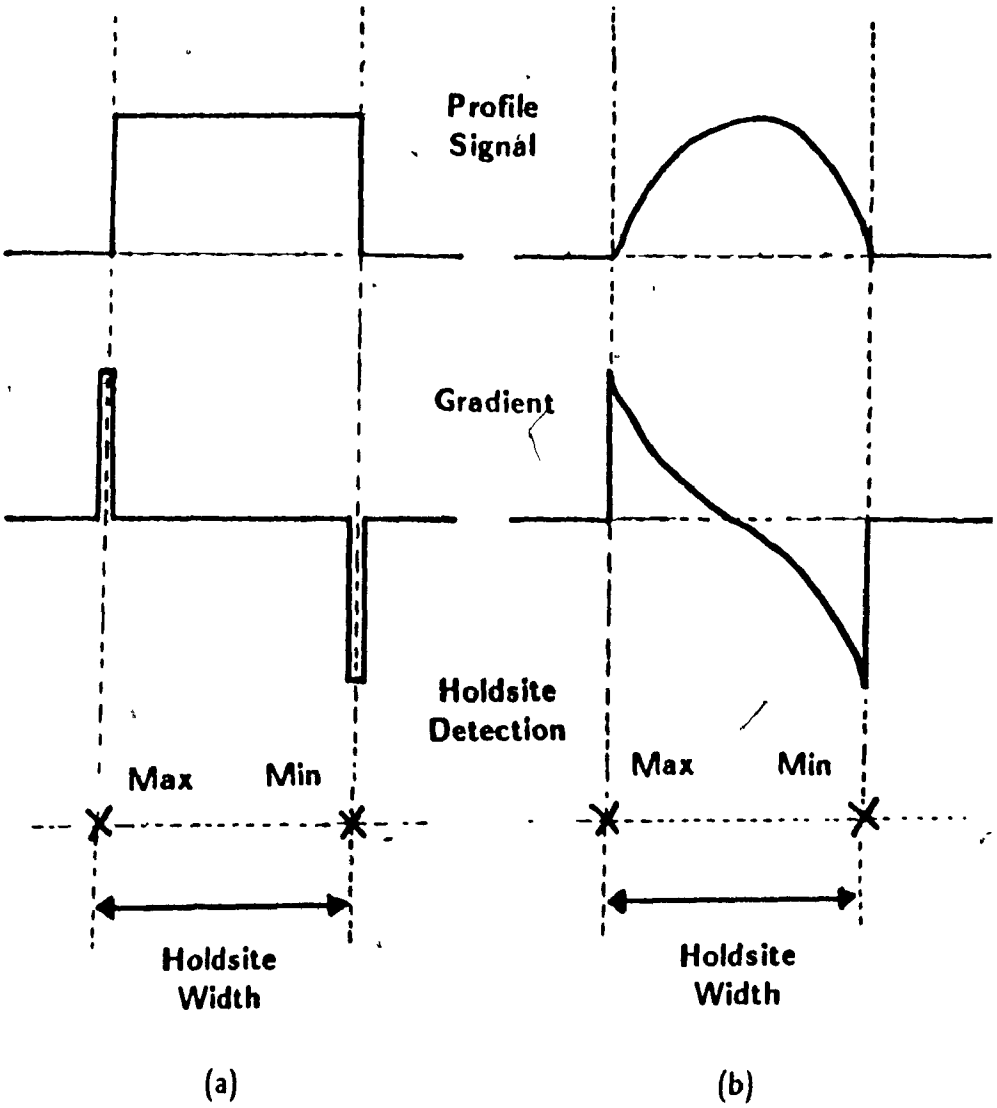


Figure 5.4 The profile processing procedure

this set consists of the locations of the clamping surfaces as detected in every individual profile. At this point we fit two lines using the least-squares approximation, one through $\{z(j)\}$ and another through $\{z(j_p)\}$, as indicated by equations (5.1), (5.2) and (5.3). The purpose of this procedure is to combine the results of profile processing so as to obtain a 3-D representation of the holdsite.

Let

$$x = ay + b \tag{5.1}$$

be the line which is approximated by a least-squares fit on the basis of a $\{x_k, y_k\}$ set of points. Then it can be shown that:

$$a = \frac{\sum y_k \sum x_k y_k - \sum y_k^2 \sum x_k}{(\sum y_k)^2 - \sum 1 \sum y_k^2} \tag{5.2}$$

$$b = \frac{\sum x_k \sum y_k - \sum 1 \sum x_k y_k}{(\sum y_k)^2 - \sum 1 \sum y_k^2} \tag{5.3}$$

If the approximated clamping surfaces are not parallel within a tolerance of ten degrees, the holdsite is rejected since this indicates a lack of consistency between the results obtained for adjacent range profiles. Otherwise the holdsite is confirmed and its location estimated as the center of gravity of the area enclosed between the two approximated lines, whereas its orientation is taken to be the average orientation of the clamping surfaces. Holdsite accessibility is computed by verifying the depth values of pixels near each of the clamping surfaces. Accessibility, as described in Section 4.4.3, is a binary parameter which

indicates whether a holdsite can be reached or not using a specific robot gripper. Figure 5.5 shows the two critical zones for accessibility. If the depth of every pixel in these zones is below the required level for object grasping, which for the gripper used in our experiments was equal to the maximum holdsite height $\max_{k,t} z_k(i)$ less 7 mm, then the holdsite is deemed accessible by the robot gripper, and a command is issued to the robot in order to attempt part acquisition. The object grasping and manipulation operations consist of several steps. First, the robot gripper is moved over the holdsite location and oriented along the holdsite's main axis, then it is moved down vertically until it reaches the already described appropriate grasping level. The robot gripper is subsequently closed so as to grab the object, and finally the object is moved to a specific location and deposited in the required orientation

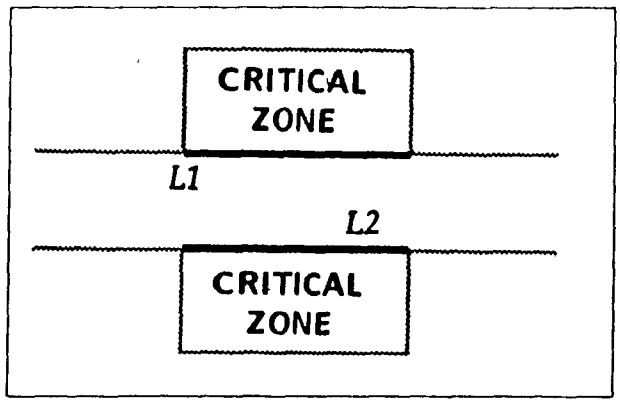


Figure 5.5 Critical zones for accessibility

We have therefore described in this section how local range information can be used as a complement to the global intensity data, in order to obtain more accurate estimates of holdsite position, orientation and accessibility.

Chapter 6 Results

In this chapter we describe the test results of the BIFOCAL binpicking system when applied to piles of cylinders and stacks of industrial parts. We also analyze the algorithm sensitivity to variations in gradient threshold values, holdsite models and amount of collected range data. Finally, timing considerations are described and suggestions given as to how to improve execution times.

6.1 Binpicking Cylinders

This section deals, through an example, with the complete procedure of picking up a cylinder out of a pile. Figure 6.1 shows an image of a pile of cylindrical objects. This intensity image is processed so as to extract the best potential holdsites, as illustrated in figure 6.2. First, the Sobel gradient of the image is computed and thresholded and lines of a pre-determined length (15 pixels) are then detected. Subsequently, potential holdsites are generated as pairs of parallel lines, and finally, these holdsites are evaluated according to their appropriateness and the three best among them are saved for further processing.

The holdsite which has the highest quality is selected as a target for grasping and the robot's end effector is moved toward it along the line-of-sight using a guarded

Figure 6.1 Pile of cylinders

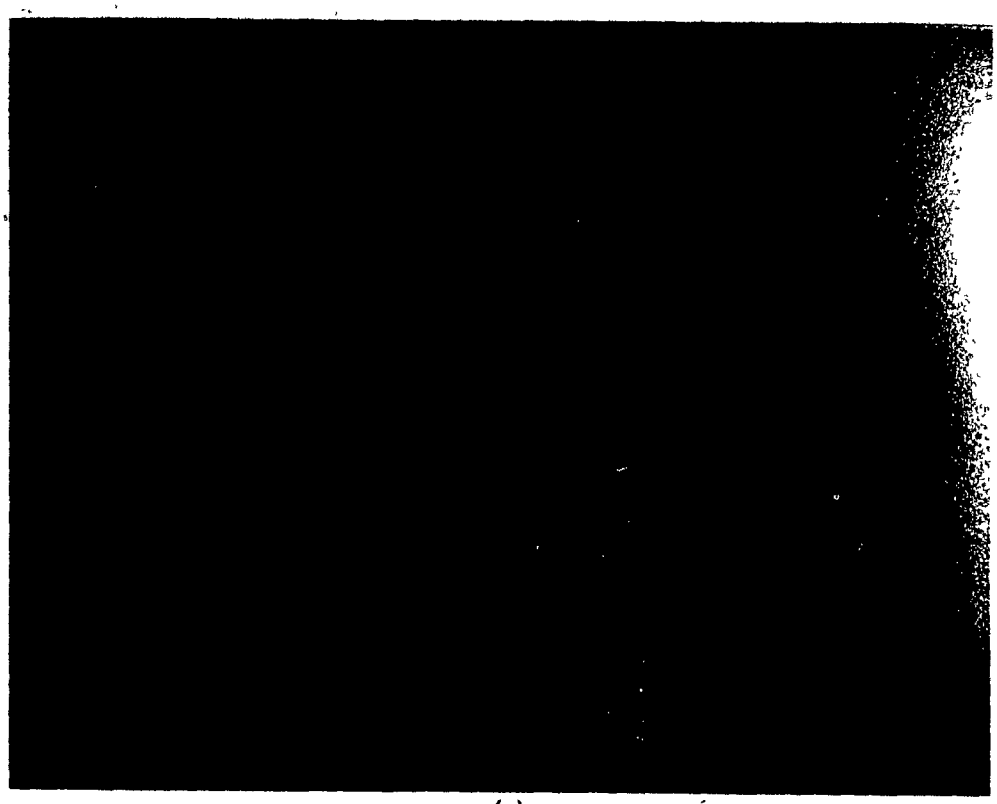
motion approach. A local range grid is then collected around the holdsite, as shown in figure 6.3. This depth map is processed in order to: confirm the presence of the holdsite, compute its exact location, and determine its accessibility by a parallel-jaw gripper. Figure 6.4 shows the local extrema of the gradient and the least-squares approximation of the holdsite's clamping surfaces. In this example the holdsite is accessible and acquisition is thus successfully attempted.

The BIFOCAL system was tested on 50 different holdsites, 40 of which were found to be accessible. Table 6.1 summarizes the results of these experiments which show a remarkable success rate of 85.0 % on the first acquisition attempt. Furthermore, considering that for a given scene three holdsites are selected, the probability of not being able to grasp any part after three consecutive acquisition attempts is very small.

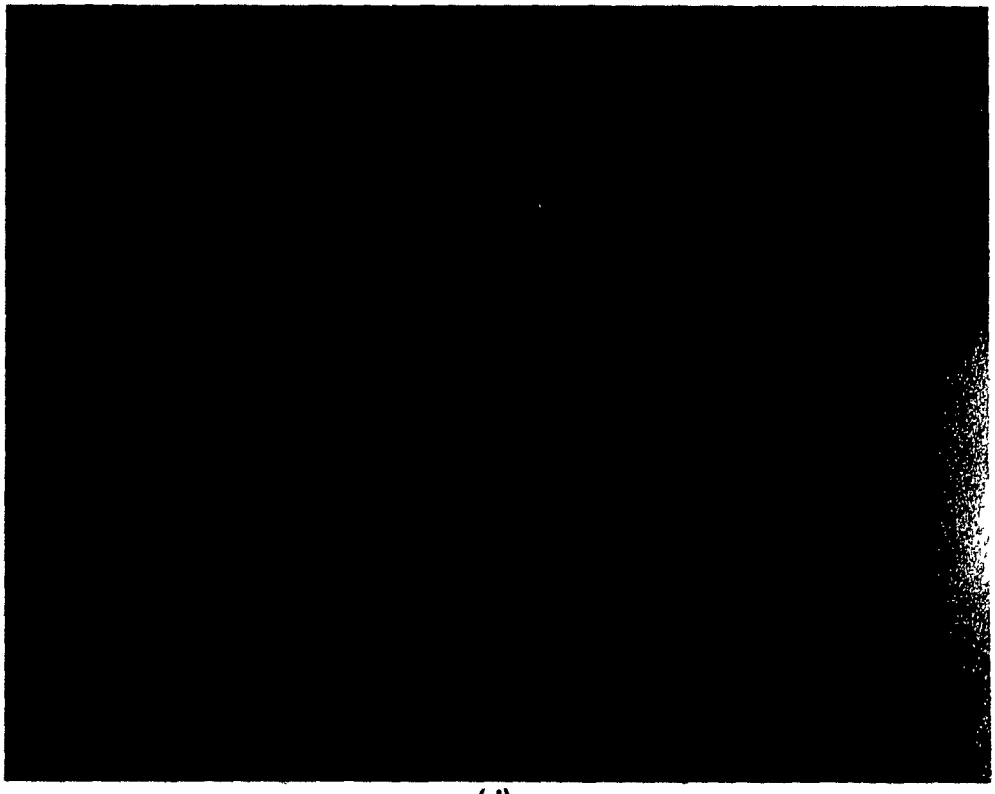
(a)

(b)

Figure 6.2 Image processing results concerning the pile of cylinders: (a) thresholded gradient and (b) line detection

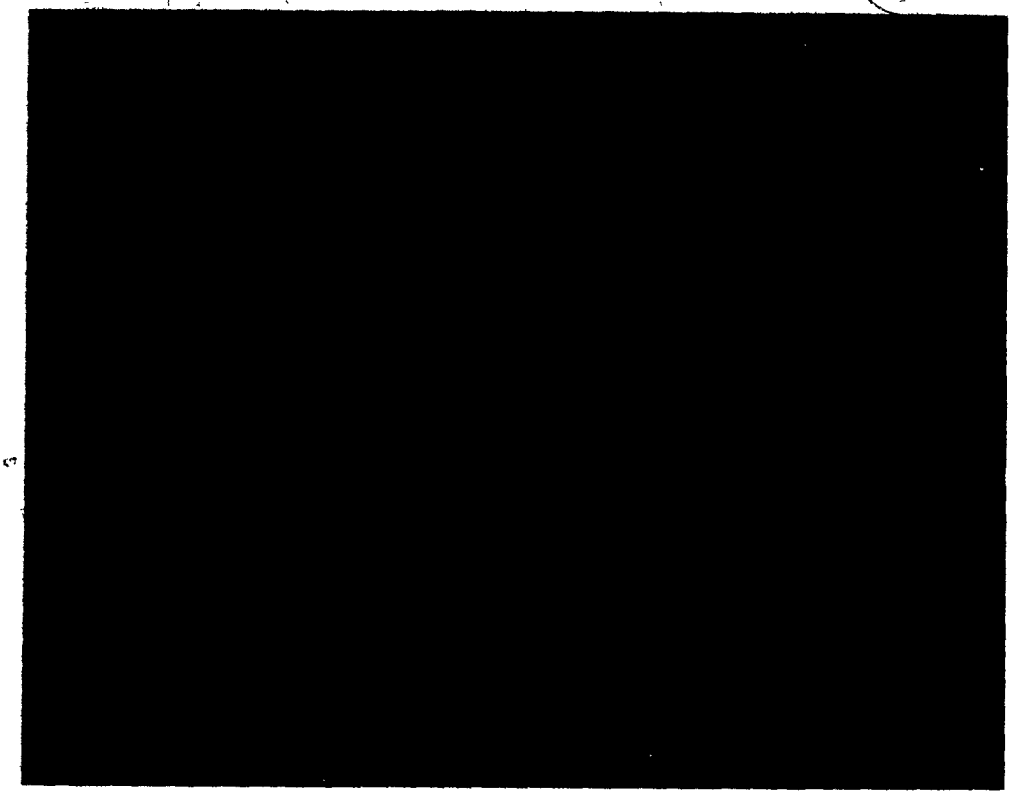


(c)

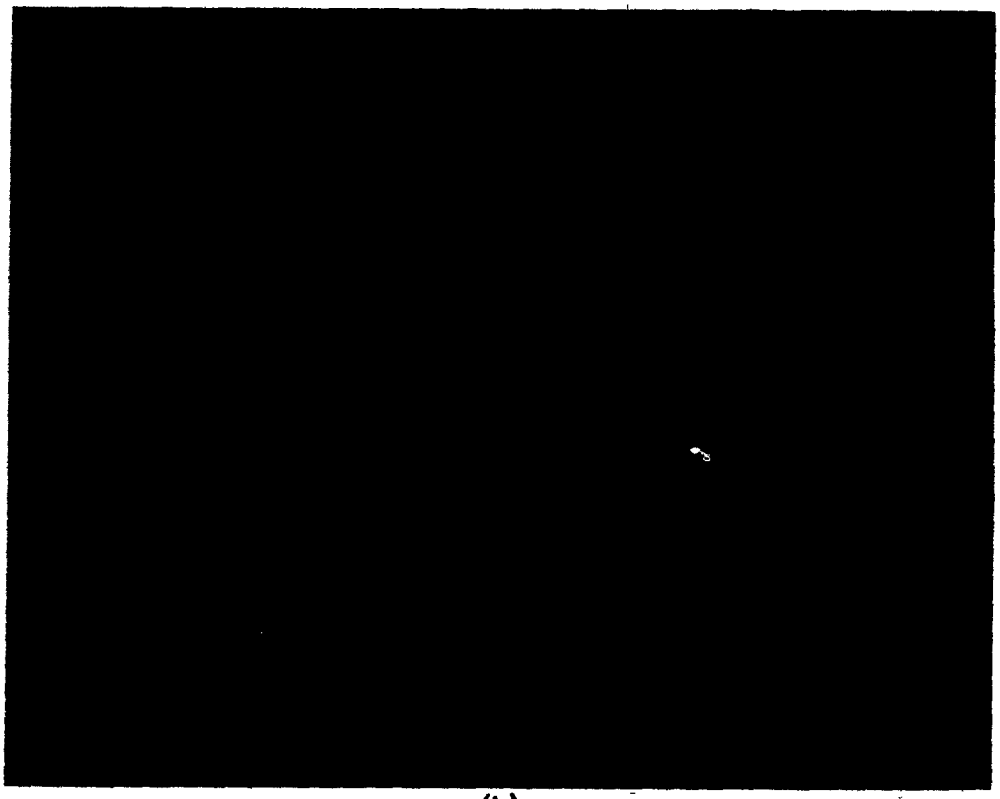


(d)

Figure 6.2 Image processing results concerning the pile of cylinders (continued): (c) holdsite detection and (d) most promising holdsites

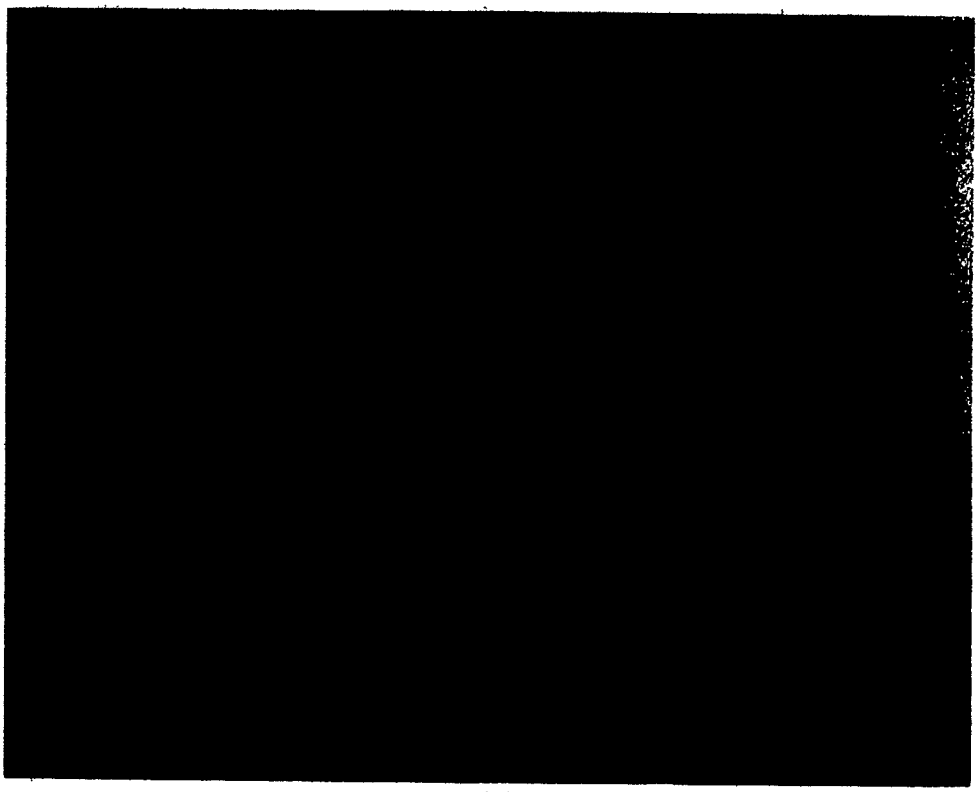


(a)

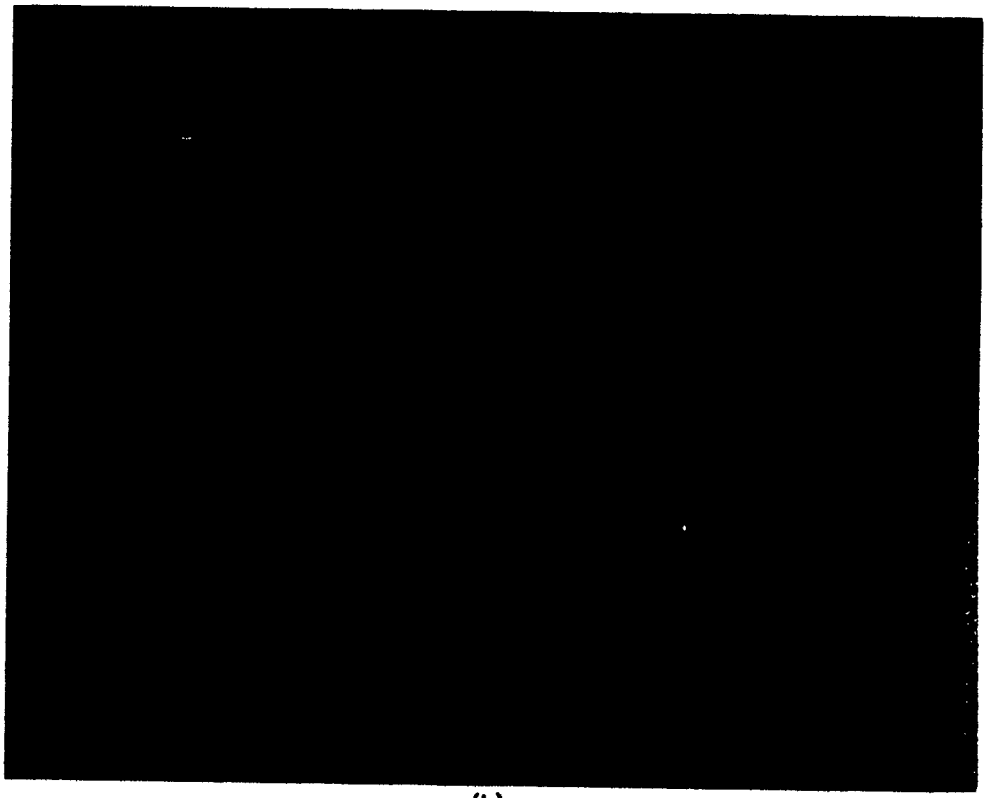


(b)

Figure 6.3 Local range grid: (a) pseudo-gray level image and (b) 3-D view



(a)



(b)

Figure 6.4 Range image processing: (a) local extrema of the first derivative and (b) least-squares approximation of the holdsite

Object Type	Total Number of Holdsites	Successfully Grasped	Failed Acquisition	Inaccessible	Success Rate
Cylinders	50	34	6	10	85.0%
Industrial Parts	50	27	8	15	77.1%
Total	100	61	14	25	81.3%

Table 6.1 Results of binpicking

6.2 Binpicking Industrial Parts

Our system was subsequently tested on piles of industrial parts, most of which were provided by General Motors of Canada Inc * while the rest were specifically designed for testing purposes Some parts had to be colored with white paint since they were too dark and could not be detected by the available range finder An example of a pile of industrial parts is given in figure 6 5 whereas figure 6 6 shows the image processing results corresponding to the particular scene

As in the previous section, the program determines the three best potential holdsites and the robot hand is sent to the most promising holdsite along the line-of-sight. However, after determining that this holdsite is inaccessible, the robot end effector is moved

* General Motors of Canada Inc P O Box 660, Ste-Thérèse, Québec Canada

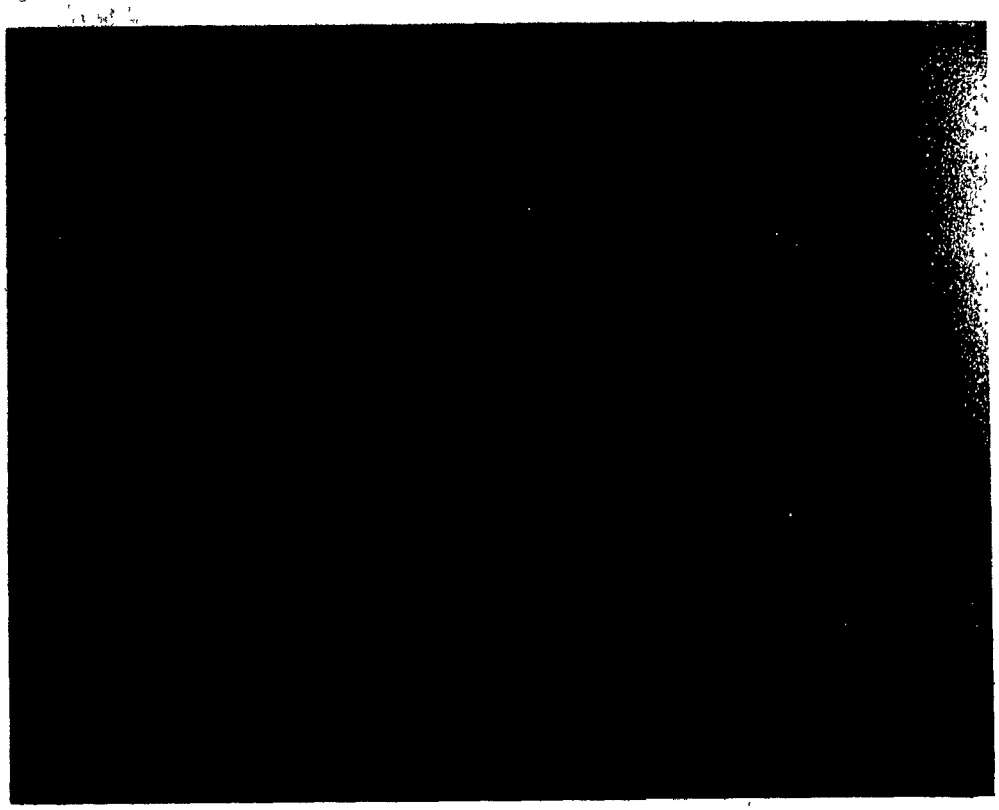


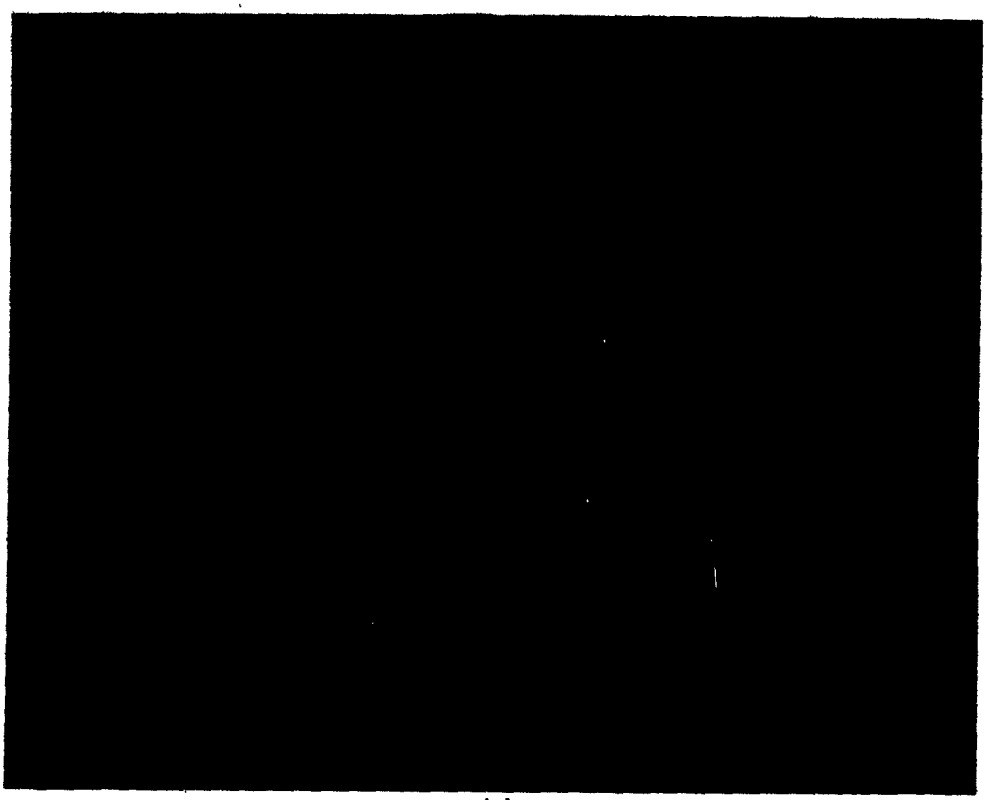
Figure 6.5 Pile of industrial parts

towards the second best holdsite A local depth grid is then collected around the holdsite, as shown in figure 6.7, and holdsite location, orientation and accessibility are computed (see figure 6 8).

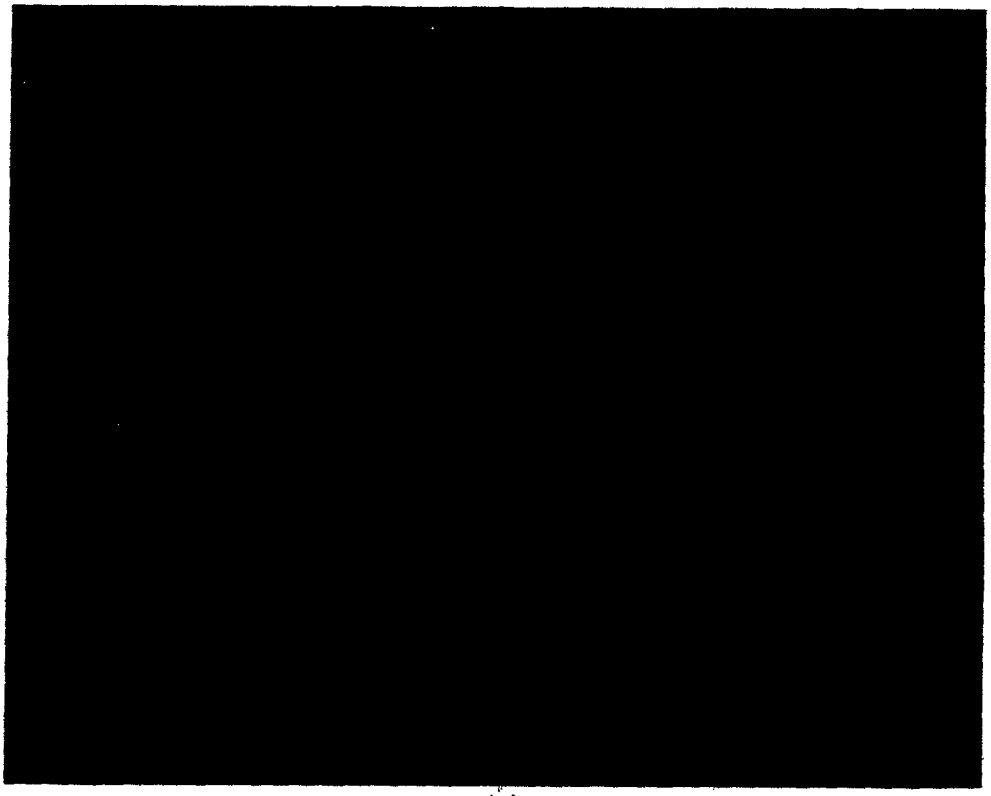
Again, we have verified the system's performance on 50 different holdsites and the results of these tests are shown in Table 6.1. It is worth noting that the BIFOCAL system achieved a success rate of 77.1 % on the first acquisition attempt, and this for a pile of industrial parts.

6.3 Variations in the Gradient Threshold

The thresholded gradient image constitutes the basic input to the line-finding algorithm and is therefore critical to the success of the complete 2-D holdsite detection



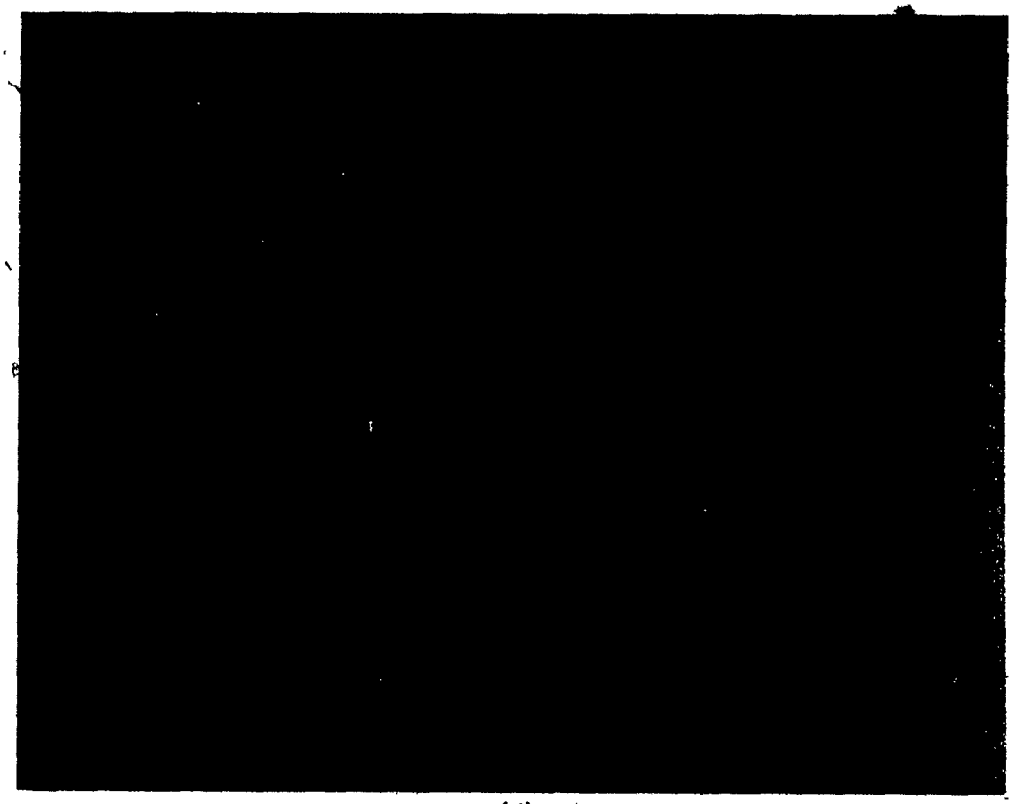
(a)



(b)

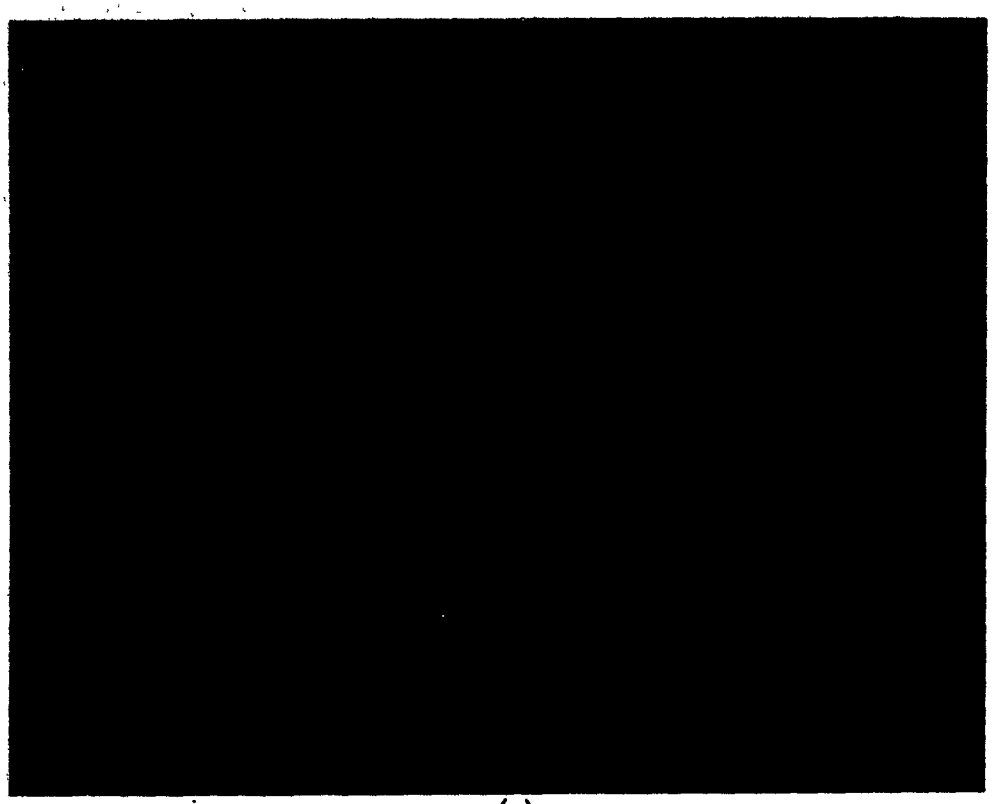
Figure 6.6 Image processing results concerning the pile of industrial parts: (a) thresholded gradient and (b) line detection

(c)

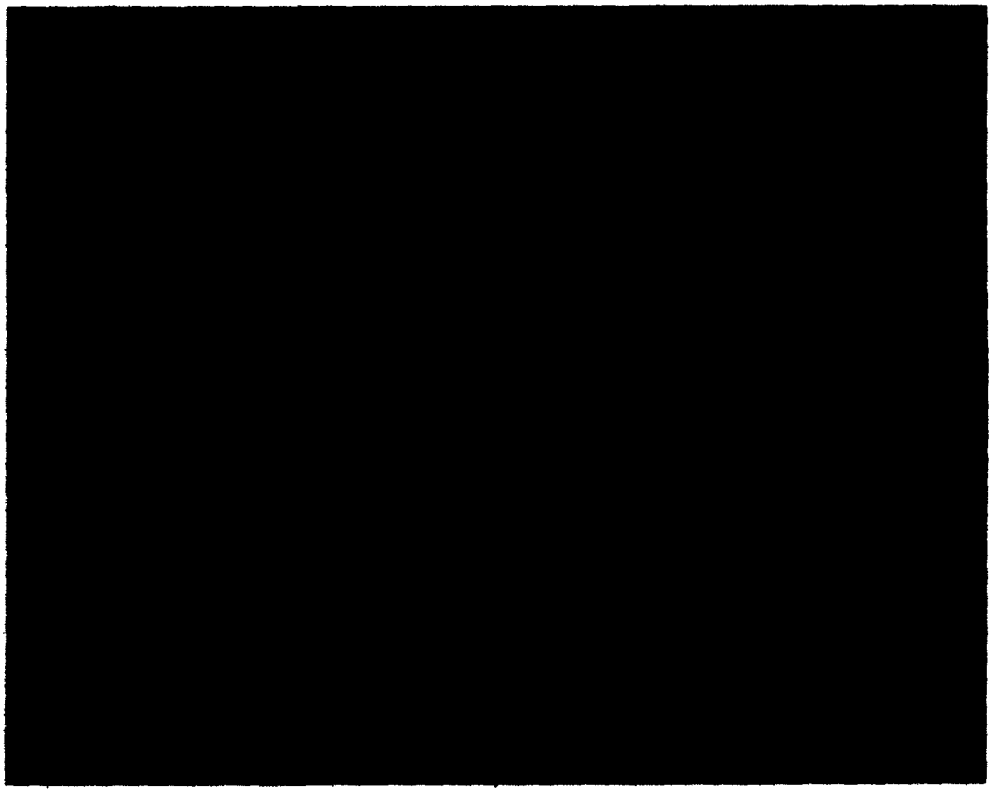


(d)

Figure 6.6 Image processing results concerning the pile of industrial parts (continued): (c) holdsite detection and (d) most promising holdsites



(a)



(b)

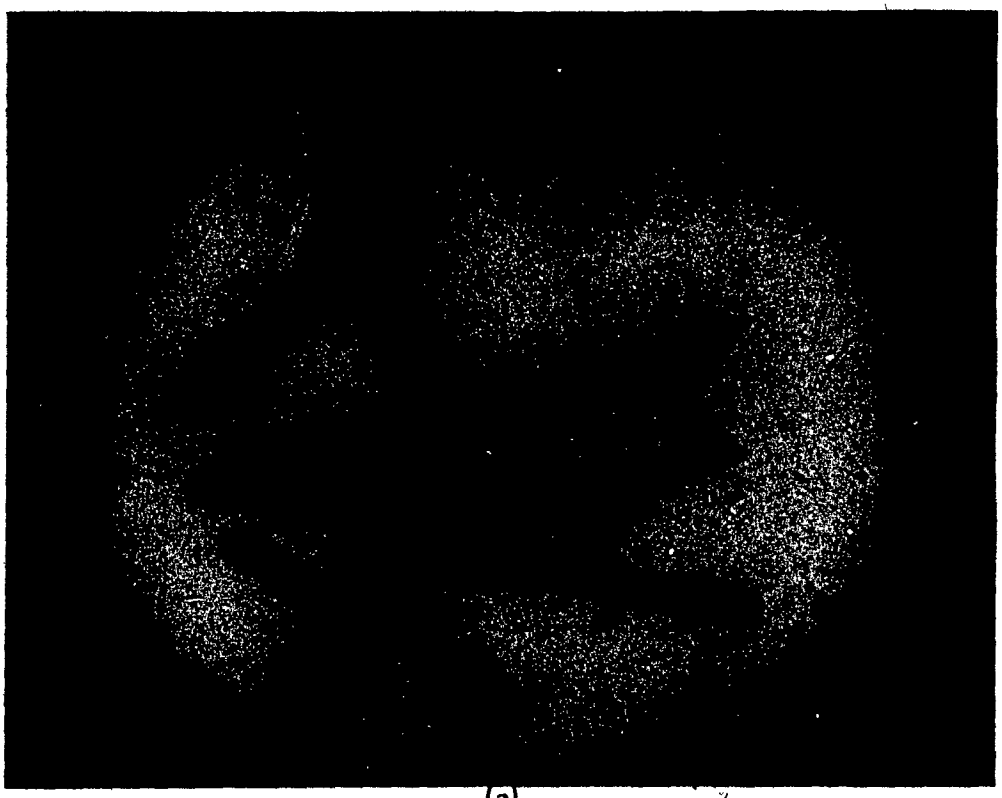
Figure 6.7 Local range grid: (a) pseudo-gray level image and (b) 3-D view

(a)

(b)

Figure 6.8 Range image processing: (a) local extrema of the first derivative and (b) least-squares approximation of the holdsite

procedure. In this section we study the sensitivity of the holdsite finding algorithm to variations in the value of the gradient threshold. Figures 6.9 and 6.10 show the detected potential holdsites for different gradient thresholds in the respective cases of a pile of cylinders and a stack of industrial parts.



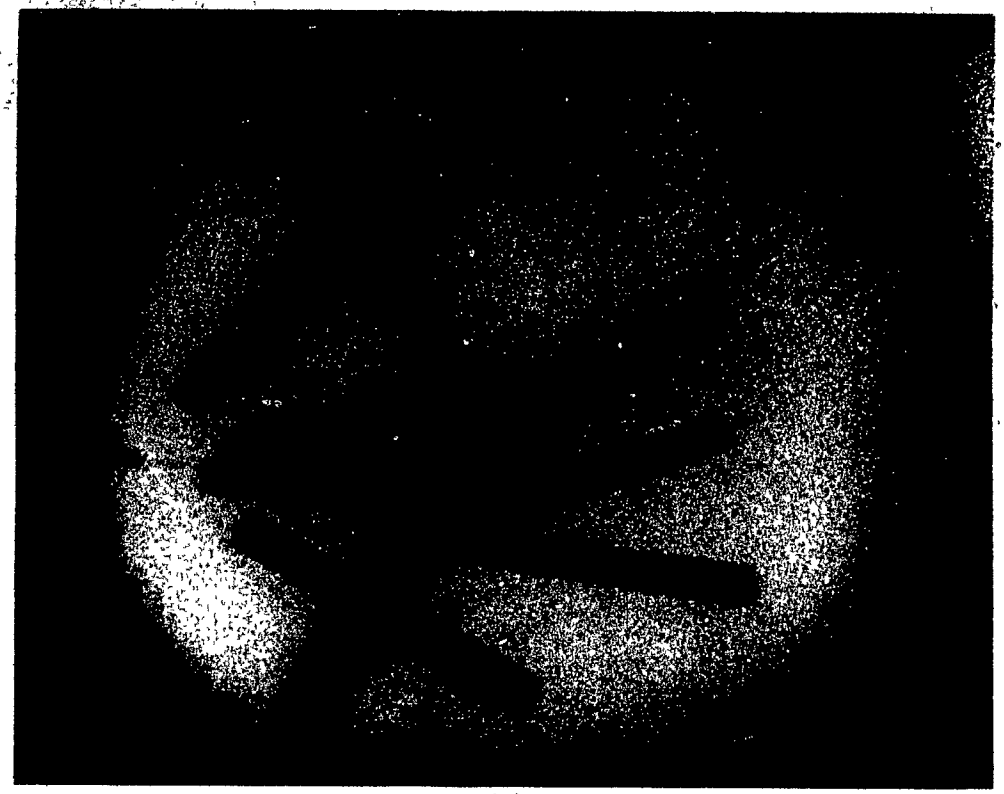
(a)

Figure 6.9 Computed holdsites for a pile of cylinders (a) threshold = 5

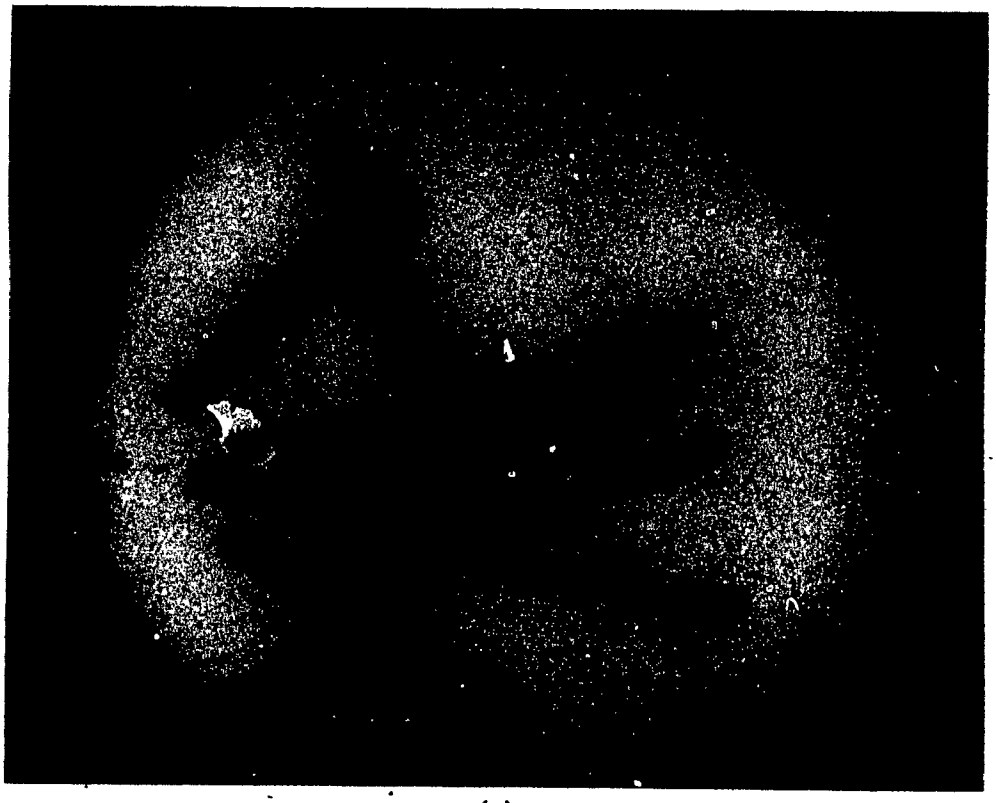
These results indicate that there is a wide range of threshold values for which the program performs adequately. In other words, the holdsite finding algorithm is robust as far as gradient threshold variations are concerned, and it therefore has a desirable low sensitivity to lighting conditions.

6.4 Variations in the Holdsite Model

The holdsite model, as previously described, consists of only two parameters:



(b)



(c)

Figure 6.9 Computed holdsites for a pile of cylinders (continued)· (b) threshold = 15 and (c) threshold = 25

(a)

(b)

Figure 6.10 Computed holdsites for a pile of industrial parts: (a) threshold = 5 and (b) threshold = 15

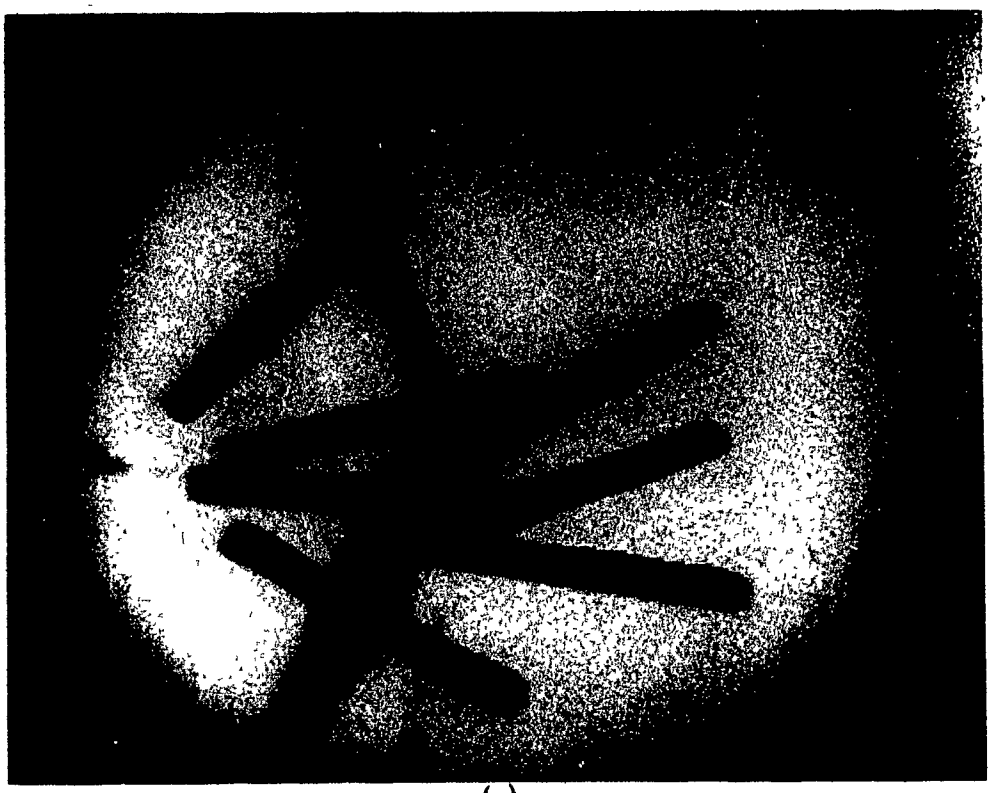
(c)

Figure 6.10 Computed holdsites for a pile of industrial parts (continued) (c) threshold = 25

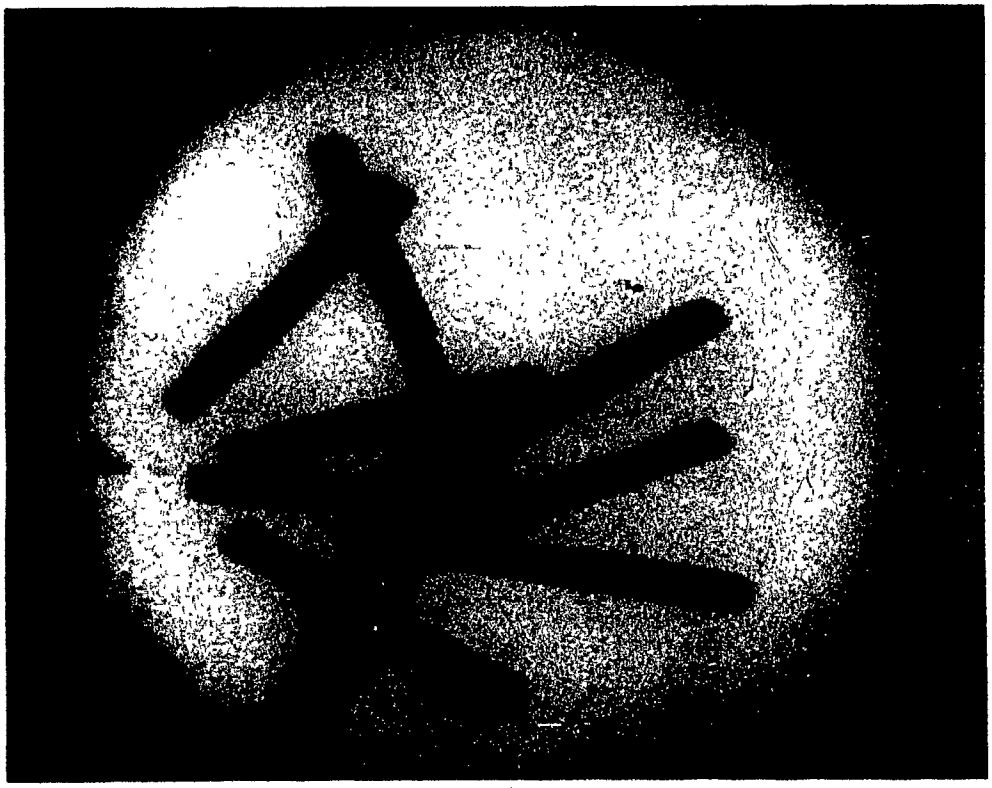
width and length. Variations in the values of any of these two parameters are bound to have an effect on the algorithm's performance. Figures 6.11 and 6.12 illustrate the sensitivity of the holdsite detecting procedure to variations in the width of the holdsite model.

As expected, significant variations in the width value considerably alter the detection of potential holdsites. This is a desirable property, since the width parameter acts as a holdsite filter, discarding those parallel lines that are too close together or too far apart to correspond to a legal holdsite according to the model. However, we note that holdsite detection is not sensitive to small variations (of the order of 20 %) in the value of the model width.

Thus we observe that the BIFOCAL system is flexible in that it disregards small width variations, while being sensitive enough to be able to sort parts according to their

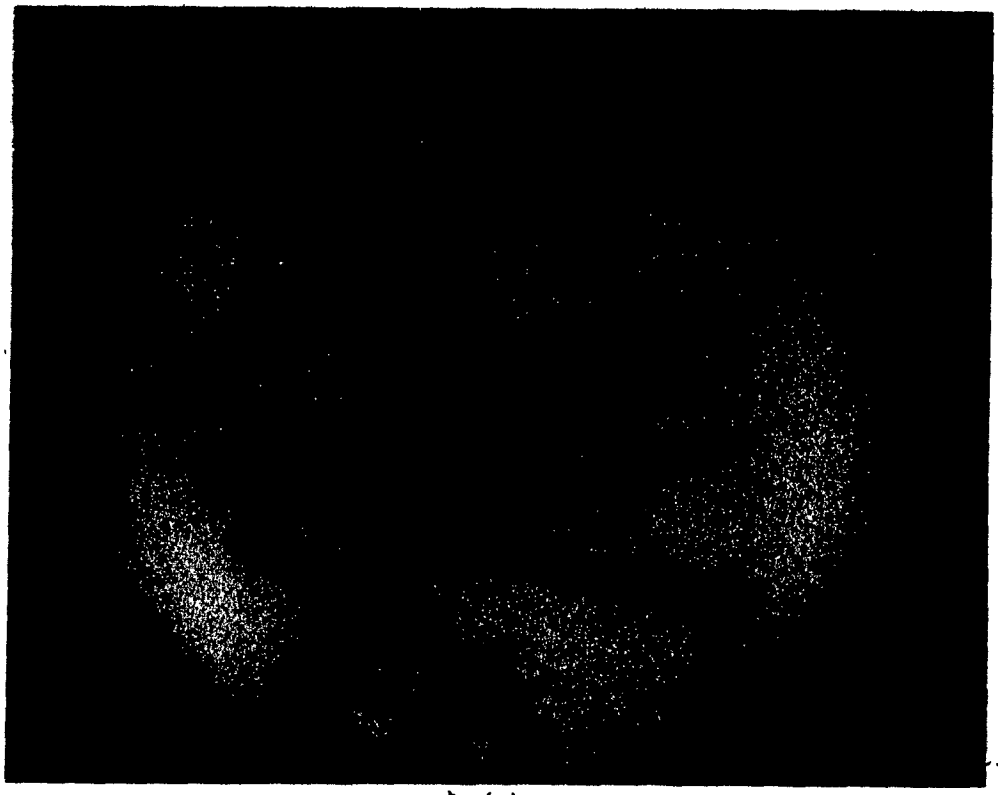


(a)

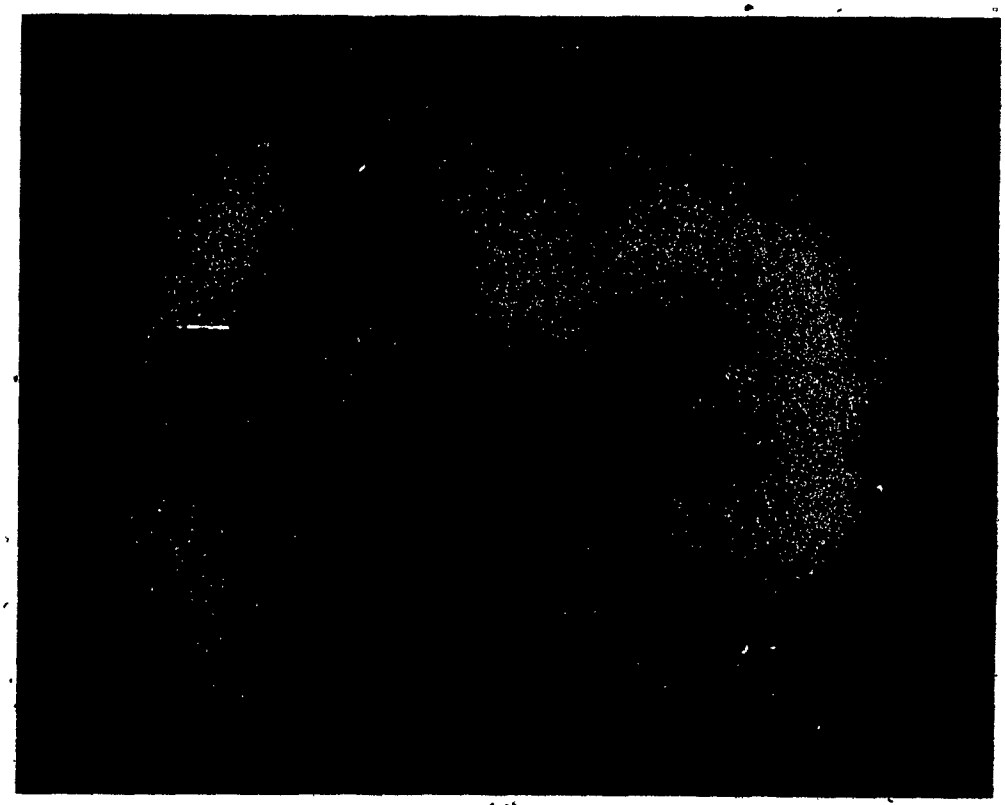


(b)

Figure 6.11 Computed holdsites for a pile of cylinders. (a) width = 9 mm and (b) width = 11 mm

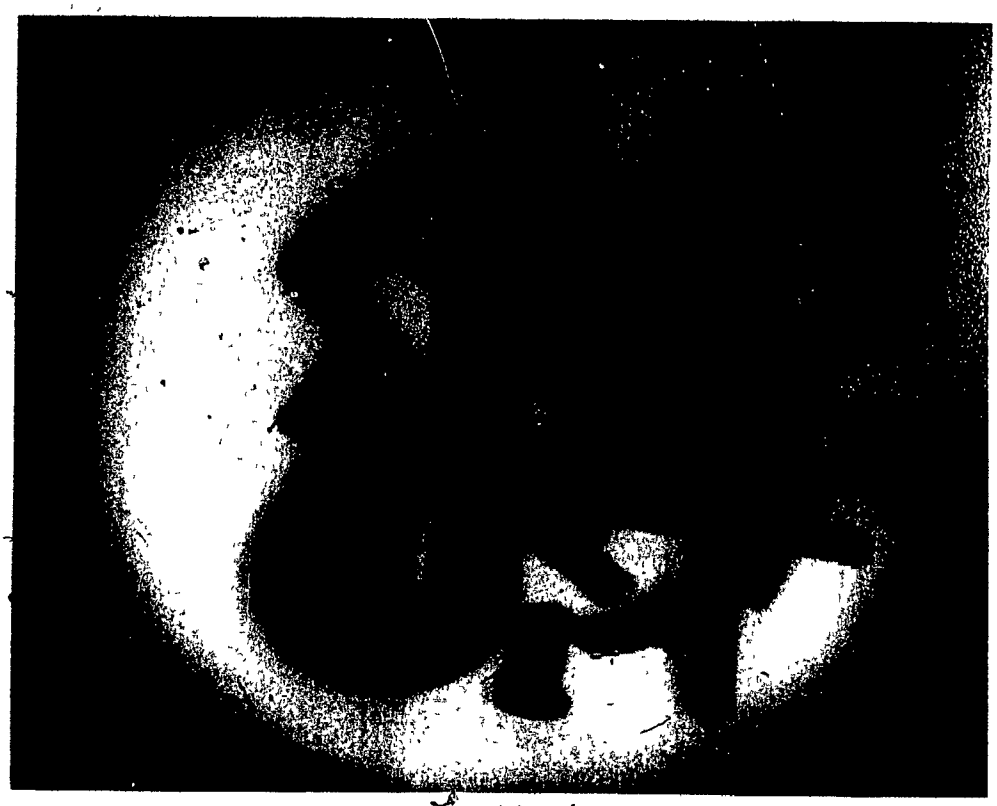


(c)

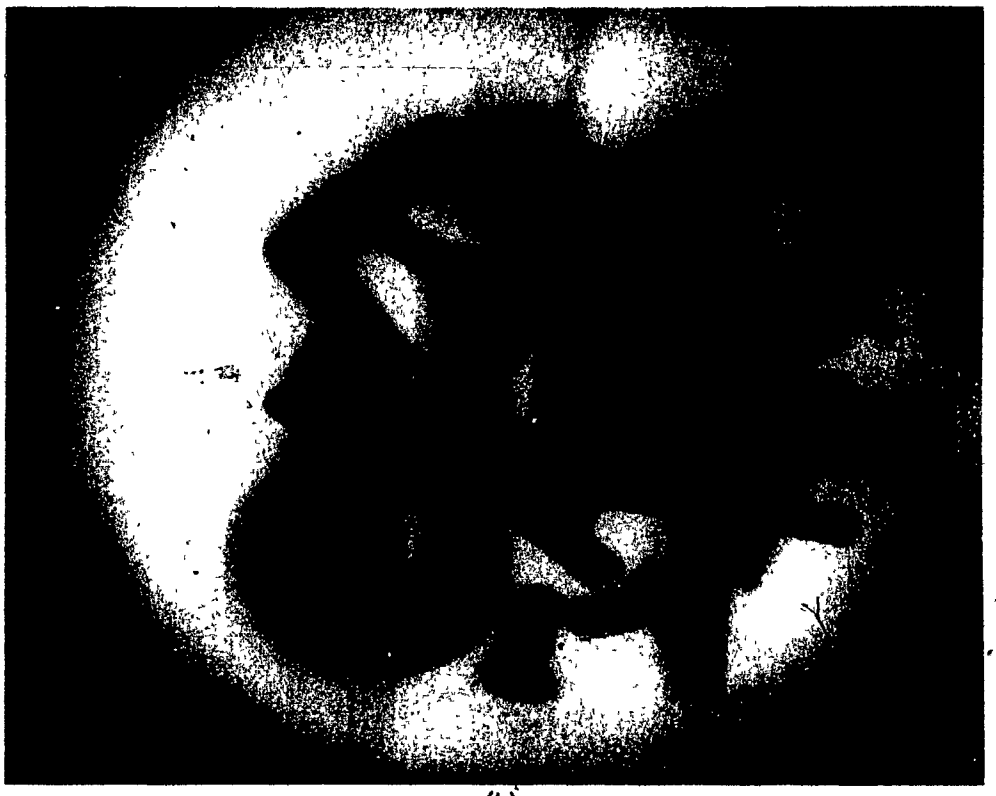


(d)

Figure 6.11 Computed holdsites for a pile of cylinders (continued): (c) width = 13 mm and (d) width = 15 mm

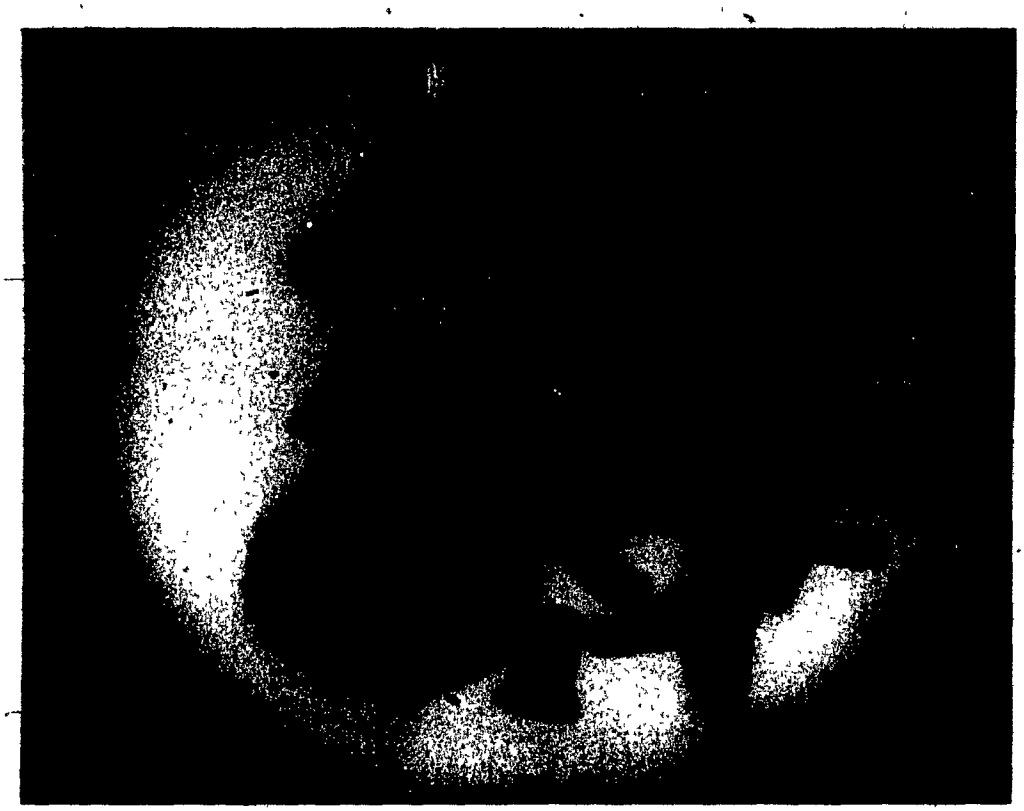


(a)

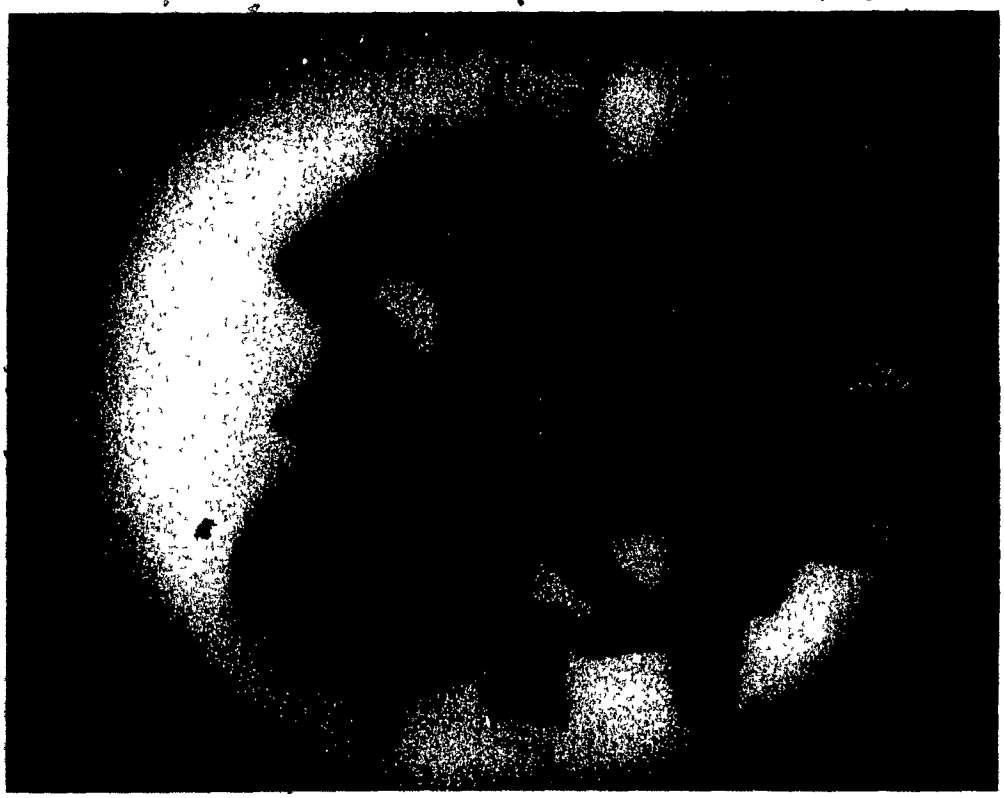


(b)

Figure 6.12 Computed holdsites for a pile of industrial parts: (a) width = 9 mm and (b) width = 11 mm



(c)

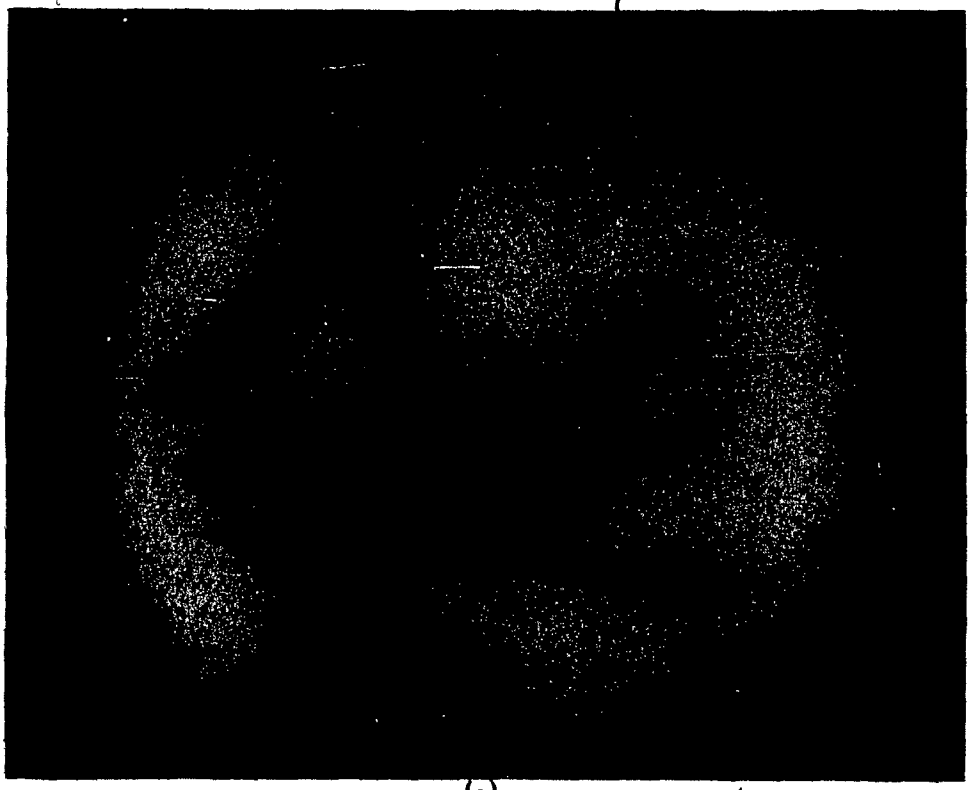


(d)

Figure 6.12 Computed holdsites for a pile of industrial parts (continued): (c) width = 15 mm and (d) width = 17 mm

size.

Figures 6.13 and 6.14 show computed holdsites as a function of the length of the holdsite model. These results vary gracefully for different values of model length. In other words, there is a large number of length values which result in adequate system performance.

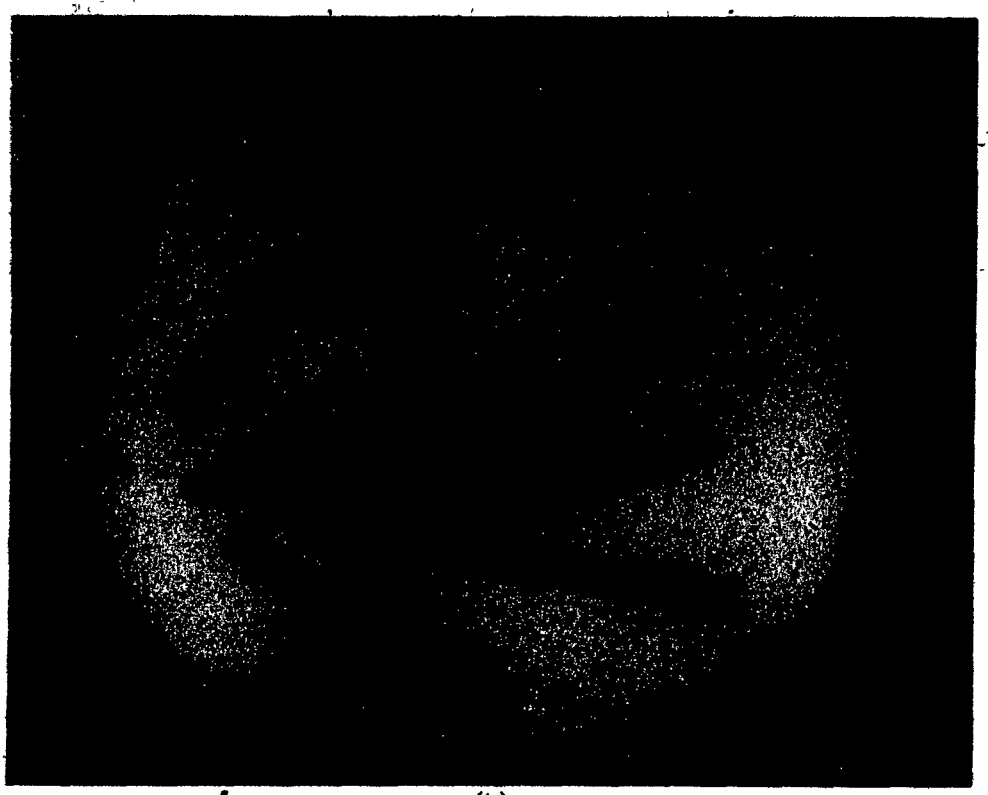


(a)

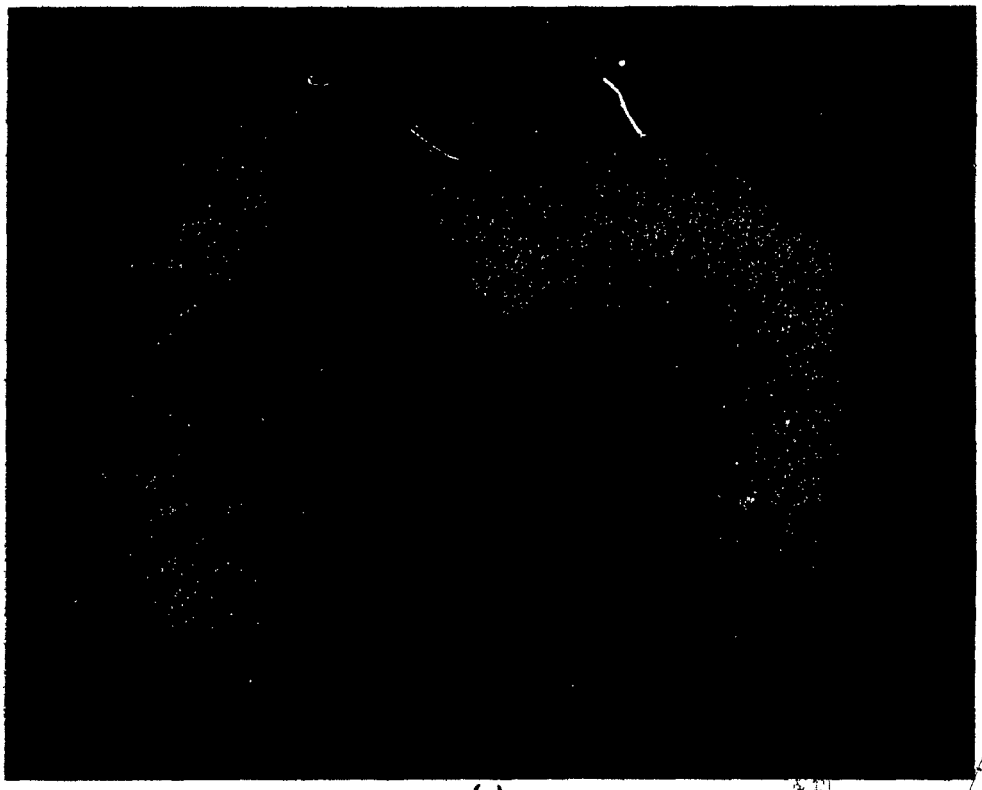
Figure 6.13 Computed holdsites for a pile of cylinders. (a) length = 15 mm

6.5 Variations in the Amount of Collected Range Data

In this section, we analyze the sensitivity of the range image processing algorithm to variations in the amount of collected depth data. To this end we performed a series of tests which consisted of measuring the error in the computed location and orientation of a cylinder, as a function of the number of collected scan lines (i.e. range profiles).



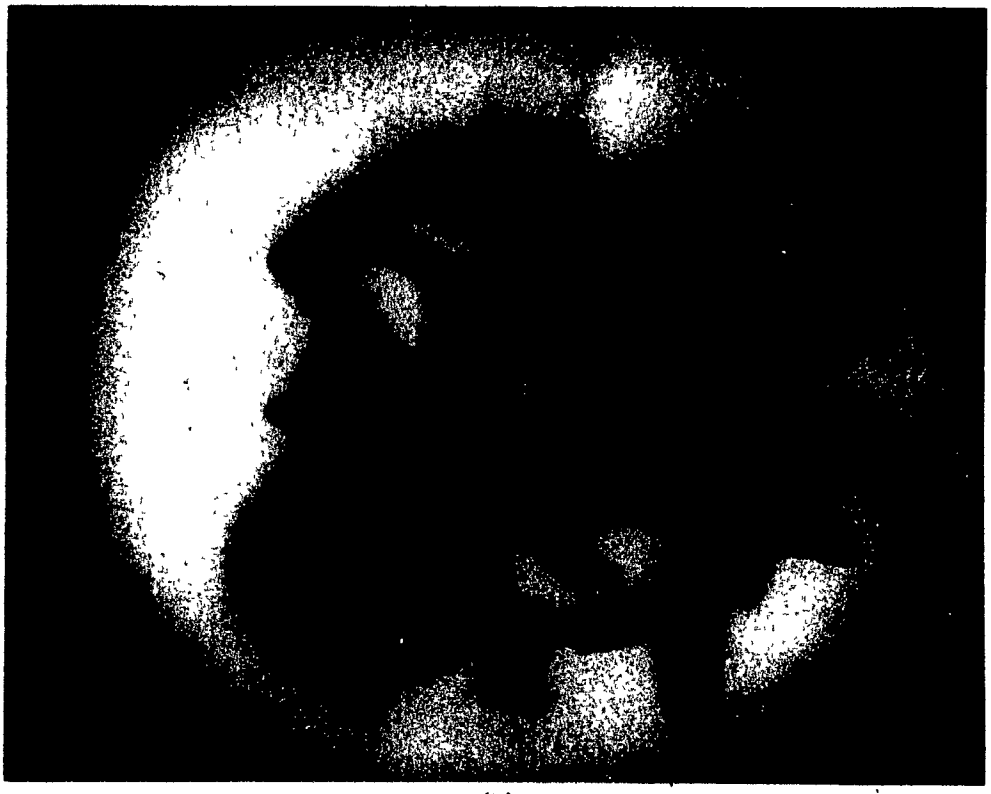
(b)



(c)

Figure 6.13 Computed holdsites for a pile of cylinders (continued): (b) length = 30 mm and (c) length = 40 mm

(a)



(b)

Figure 6.14 Computed holdsites for a pile of industrial parts: (a) length = 15 mm and (b) length = 30 mm

(c)

Figure 6.14 Computed holdsites for a pile of industrial parts (continued) (c) length = 40 mm

The results of these tests are illustrated in figures 6.15 and 6.16, and show that both the location and the orientation errors decrease as the number of scan lines increases.

The higher accuracy brought about by the addition of range data is due to the fact that the extra information increases redundancy and diminishes the statistical probability of error. However, the ideal number of scan lines must be a compromise between scanning speed on one side, and accuracy of the computed holdsite location, orientation and accessibility, on the other side. Furthermore, the absolute errors in holdsite location and orientation estimates are, in our context, very small, even in the case of only *two* scan lines. Because of this we chose to collect four profiles per grid, which in our view constitutes an appropriate trade-off between speed and computational precision. We therefore conclude that, due to the high quality of the acquired range data, the BIFOCAL system is relatively

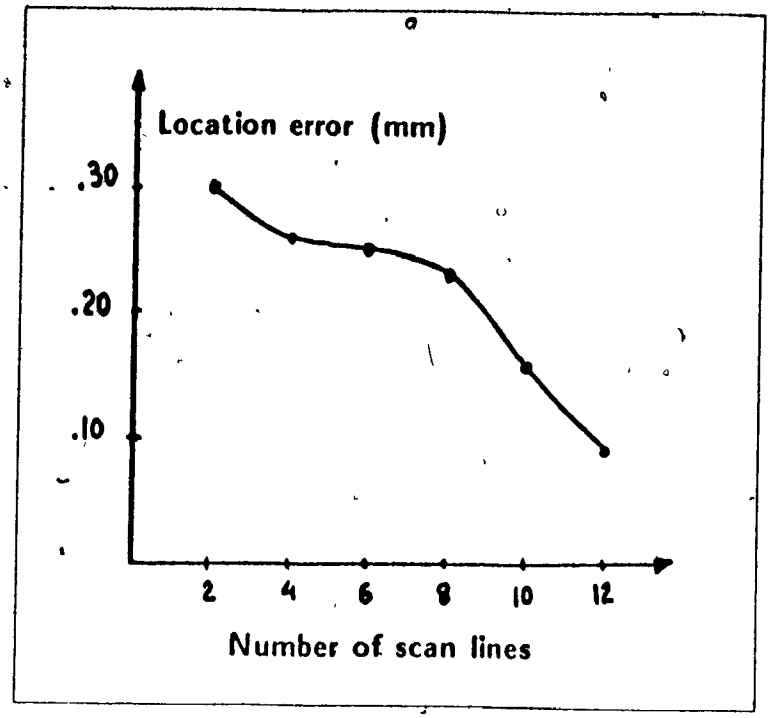


Figure 6.15 Holdsite location vs no of scan lines

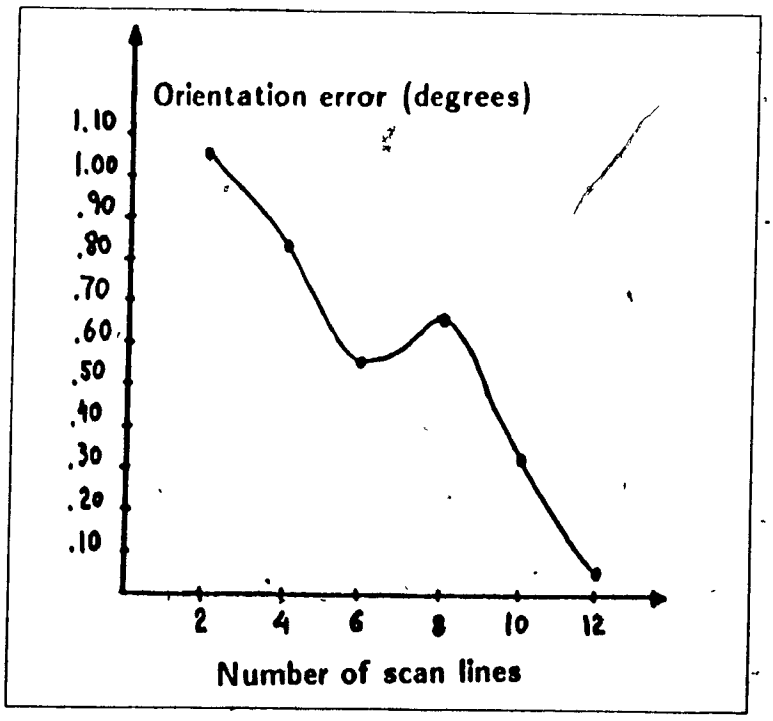


Figure 6.16 Holdsite orientation vs. no of scan lines

insensitive to variations in the amount of collected range data.

6.6 Timing Considerations

Under its current implementation, the BIFOCAL system takes an average of four minutes to acquire and manipulate a part from a pile. The approximate breakdown of the execution time is as follows:

- T.V. image capture: 55 s.
- Sobel gradient computation and thresholding: 35 s.
- Line finding: 45 s.
- Holdsite detection: 15 s.
- Line-of-sight approach: 20 s.
- Fine approach and gripper positioning: 15 s.
- 3-D scanning (consisting of four profiles): 30 s.
- Range image processing: 5 s.
- Object acquisition: 10 s.
- Object manipulation: 10 s.

It is worth noting that if we reduced the unnecessary overhead associated with image capture (i.e. almost one minute) * and we optimize the image processing algorithms as well as the robot motions, we could easily achieve execution times of approximately two minutes per object. Moreover, if all of the processing algorithms were implemented in the form of VLSI circuits, namely the Sobel gradient computation and thresholding, line finding, holdsite detection and range image processing, the only limiting factor would then be robot speed. Therefore, we consider that there is significant room for improvement in the area of execution speed, particularly if this system is to be developed into an industrial product.

* This image acquisition delay is due to the fact that, in our experimental set-up, the image must be sent through the computer network before it can be processed.

Chapter 7 Conclusion

The problem of binpicking is a relevant one in industry today. Many researchers have attempted to find a general solution but, in general, they have ended up developing context dependent algorithms . In this thesis we have described the BIFOCAL system whose purpose is to acquire and manipulate three-dimensional objects which are initially piled up and randomly oriented. The only restriction on parts is that they must have at least one appropriate holdsite that can be grasped by a parallel-jaw gripper.

The binpicking algorithm is holdsite-driven, in other words, no attempt is made to recognize object identity. Only holdsites are of interest since any part may be grasped out of the pile. One of the system's main contributions is the fact that two sources of sensory input are used so that they complement each other: a T.V. camera placed over the workspace captures intensity images of the global scene, and a wrist-mounted single-point range finder collects local 3-D data. The brightness images are used to find the location of the potential holdsites and to evaluate their quality on the basis of appropriate suitability criteria. At this point, the robot hand is moved so that the range finder can collect a local grid of 3-D data around the most promising holdsite. Range data are used to confirm the presence of the holdsite, compute its exact location and orientation, and determine its

accessibility by the robot gripper. These local data contain accurate information about the geometry of the holdsite and its neighborhood, and thus provide the system with a much higher degree of robustness and reliability than can be found in previous systems. Furthermore, we only used inexpensive, commercially available sensors, as opposed to custom made range finders

The flexibility of the system has been experimentally confirmed, as it performed consistently well when faced with substantial variations in the values of the gradient threshold, the holdsite model and the amount of collected range data. We have therefore proven the feasibility and appropriateness of a holdsite-based system that integrates 2-D and 3-D inputs and that provides reliable binpicking of 3-D objects. Future work should concentrate on improving the execution speed of the overall program, as well as explore alternative range image processing techniques. In this regard, the replacement of the single-point range finder by a more adequate 3-D sensor (i.e. with a larger measuring range) would certainly have a positive effect on the system's performance.

References

[1] G. J. Agin, M. J. Uram, and P. T. Highnam, "Three-dimensional sensing and interpretation", Technical report CMU-RI-TR-85-1, Carnegie-Mellon University, The Robotics Institute, Jan 1985

[2] C Archibald and M. Rioux, "Witness a system for object recognition using range images", Technical report ERB-986, NRCC no 25588, Jan 1986

[3] N Ayache and O D Faugeras, "HYPER a new approach for the recognition and positioning of two-dimensional objects", IEEE Trans Pattern Analysis and Machine Intelligence, vol PAMI-8, no 1, pp 44-54, Jan 1986

[4] D H Ballard, "Generalizing the Hough transform to detect arbitrary shapes", Pattern Recognition, vol 13, pp 111-122, 1981

[5] D. A. Bell, "Decision trees, tables, and lattices", Pattern Recognition, pp 119-141, New York Plenum, 1978.

[6] B Bhanu and O. D. Faugeras, "Shape matching of two-dimensional objects", IEEE Trans Patt. Anal Mach. Intell., vol. PAMI-6, pp. 137-156, March 1984

[7] J. D. Boissonnat, "Stable matching between a hand structure and an object silhouette", IEEE Trans. Patt Anal. Mach. Intell , vol. PAMI-4, no. 6, pp 603-612, Nov 1982.

[8] R. C. Bolles, "Robust feature matching through maximal cliques", Proc. SPIE Tech. Symp. Imaging Appl. Automated Industrial Inspect. Assembly, Bellingham, Wash.: Society of Photo-Optical Instrumentation Engineers, Apr. 1979.

[9] R. C. Bolles and R. A. Cain, "Recognizing and locating partially visible objects: the local-feature-focus method", The International Journal of Robotics Research, vol. 1, no 3, pp. 57-82, Fall 1982.

[10] R C Bolles and P Horaud, "3DPO A three-dimensional part orientation system", The Int Journ of Robotics Research, vol 5, no 3, pp 3-26, Fall 1986

[11] L S Davis, "Shape matching using relaxation techniques", IEEE Trans Patt Anal Mach Intell, vol PAMI-1, pp 60-72, Jan 1979

[12] J D Dessimoz, J R Birk, R B Kelley, H A Martins, and Chi Lin I, "Matched filters for bin picking", IEEE Trans Patt Anal Mach Intell., vol. PAMI-6, no. 6, pp 686-697, Nov 1984

[13] M Driels, M. Huang, R. Liscano, and K. Michael, "The use of visual feedback for the acquisition of pseudorandomly oriented parts", Journal of Robotic Systems, 1(2), 195-204, 1984.

[14] S A. Dudani, K J Breeding, and R. B. McGhee, "Aircraft identification by moment invariants", IEEE Trans Computers, vol. C-26, pp. 39-46, Jan. 1977.

[15] A. Ferloni, I. Franchetti, P. Vincentini and P. Fici, "Ordinatore: a dedicated robot

that orientates objects in a predetermined direction", Proc. 10th Int. Symp. on Industrial Robots, pp. 655-658, Milan, Italy, 1980.

[16] Y. Fukada, H. Doi, K. Nagamine, and T. Inari, " Relationships-based recognition of structural industrial parts stacked in a bin", Robotica, vol. 2, pt. 3, pp. 147-154, July 1984

[17] G. J Gleason and G J Agin, "A modular vision system for sensor-controlled manipulation and inspection", Proc. 9th Int Symp Industrial Robots, pp 57-70, 1979

[18] W E Grimson and T. Lozano-Perez, "Model-based recognition and localization from sparse range or tactile data", M I.T Cambridge, MA, A I. memo 738, Aug 1983.

[19] W. E Grimson, "Sensing strategies for disambiguating among multiple objects in known poses", MIT A.I. memo 855, 35 pages, Aug. 1985.

[20] W. Hattich, "Recognition of overlapping workpieces by model directed construction of object contours", in Artificial Vision for Robots, I. Aleksander, Ed. Chapman and Hall, N. York, pp. 77-92, 1984.

[21] B. K. Horn and K. Ikeuchi, "The mechanical manipulation of randomly oriented parts", Scientific American, pp. 100-111, Aug. 1984.

[22] M. Hu, "Visual pattern recognition by moment invariants", IRE Trans. Inform. Theory, vol. IT-8, pp. 179-187, Feb. 1962.

[23] M. H. Hueckel, "A local visual operator which recognizes edges and lines", J. ACM, vol. 20, no. 4, pp. 634-647, 1973; also see M. H. Hueckel, Erratum, J. ACM, vol. 21, no. 2, p. 350, 1974.

[24] K. Ikeuchi, H. K. Nishihara, B. K. Horn, P. Sobalvarro, and S. Nagata, "Determining grasp configurations using photometric stereo and the PRISM binocular stereo system", The Int. Journ. Robotics Research, vol. 5, no. 1, pp. 46-65, Spring 1986.

[25] L Kanal, "Patterns in pattern recognition 1968-1974", IEEE Trans Inform. Theory, vol IT-20, pp. 697-722, Nov 1974

[26] R B. Kelley, H A Martins, J R Birk and J D Dessimoz, "Three vision algorithms for acquiring workpieces from bins", Proceedings of the IEEE, vol. 71, no. 7, pp. 803-820, July 1983.

[27] R. B. Kelley, "Heuristic vision algorithms for bin-picking", Proc. 14th Int. Symp. Industrial Robots and 7th Int. Conf. Industrial Robot Technology, pp. 599-610, Sweden, Oct. 1984.

[28] T. F. Knoll and R. C. Jain, "Recognizing partially visible objects using feature indexed hypotheses", IEEE Journal of Robotics and Automation, vol. RA-2, no. 1, pp. 3-13, March 1986.

[29] M. W. Koch and R. L. Kashyap, "A vision system to identify occluded industrial parts", 1985 IEEE Int. Conf. on Robotics and Automation, pp. 55-60, March 1985.

[30] M. D. Levine. "Vision in man and machine". Mc Graw Hill, 1986.

[31] J. Lloyd. "Implementation of a robot control development environment". M. Eng. Thesis, Dept. of Electrical Engineering, McGill University, 1985.

[32] D. Lowe. "Perceptual Organization and Visual Recognition". Kluwer, Boston, 1985.

[33] D Lowe. "The viewpoint consistency constraint". Int. Journal of Computer Vision, 1, pp 57-72, 1987.

[34] A. R. Mansouri, A. S Malowany and M. D Levine. "Line detection in digital pictures: a hypothesis prediction / verification paradigm". CVGIP, vol. 40, no. 1, pp. 95-114, October 1987.

[35] J. W. McKee and J. K. Aggarwal. "Finding the edges of the surfaces of three-dimensional curved objects by computer". Pattern Recognition, vol. 7, pp. 25-52, 1975.

[36] M. Oshima and Y. Shirai. "Object recognition using three-dimensional information". IEEE Trans. Patt. Anal. Mach. Intell., vol. PAMI-5, no. 4, pp. 353-361, July 1983.

[37] W. A. Perkins. "A model-based vision system for industrial parts". IEEE Transactions on Computers, vol. C-27, no. 2, pp. 126-143, February 1978.

[38] W. A. Perkins. "Simplified model-based part locator". Proc. 5th Int. Conf. Pattern Recognition, pp. 260-263, Dec. 1980.

[39] A. Pugh. "Robot vision", Ed. by A. Pugh. University of Hull, UK, 1983.

[40] J. C. Rodger and R. A. Browse. "Combining visual and tactile perception for robotics", Proc. 6th Canadian Conf. Artificial Intell., pp. 166-171, May 1986.

[41] C. Rosen, D. Nitzan, G. Agin, A. Bavarsky and G. Gleason. "Research applied to industrial automation, 8th report", Stanford Research Institute, Menlo Park, CA, Aug. 1978.

[42] G. Roth, "Determining grasp positions for a parallel type robot gripper", NRCC technical report ERB-984, 17 pages, January 1986.

[43] G Roth and O'Hara, "A holdsite method for parts acquisition using a laser rangefinder mounted on a robot wrist", 1987 IEEE Int. Conf. on Robotics and Automation, pp. 1517-1523, April 1987.

[44] W. S. Rutkowski, S. Peleg, and A. Rosenfeld, "Shape segmentation using relaxation", IEEE Trans Patt Anal. Mach. Intell., vol. PAMI-3, pp. 368-375, July 1981.

[45] W. S. Rutkowski, "Recognition of occluded shapes using relaxation", Computer Graphics Image Processing, vol. 19, pp. 111-128, 1982.

[46] J. Segen, "Locating randomly oriented objects from partial view", SPIE Intelligent Robots; 3rd Int. Conf. on Robot Vision Sensory Controls, vol. 449, pp. 676-684, Nov. 1983.

[47] J. Serra, "Image analysis and mathematical morphology", London, UK: Academic, 1982.

[48] Y. J. Tejwani and R. A. Jones, "Machine recognition of partial shapes using feature vectors", IEEE Trans. Syst. Man, Cybern., vol. SMC-15, pp. 504-516, July 1985.

[49] J. L. Turney, T. N. Mudge and R. A. Volz, "Recognizing partially occluded parts", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-7, no. 4, pp. 410-421, July 1985.

[50] D. Van Laethem, M. Bogaert, and O. Ledoux, "A realistic approach to bin picking", Proc. SPIE Int. Soc. Opt. Engineers: Intell. Robots and Comp Vision, vol. 521, pp. 98-107, 1985.

[51] T. P. Wallace and P. A. Wintz, "An efficient three-dimensional aircraft recognition algorithm using normalized Fourier descriptors", Computer Graphics and Image Processing, vol. 13, pp. 99-126, 1980.

[52] T. P. Wallace, O. R. Mitchell, and K. Fukunaga, " Three-dimensional shape analysis using local shape descriptors", IEEE Trans. Patt. Anal. Mach. Intell., vol. PAMI-3, pp. 310-323, May 1981.

[53] H. S. Yang and A. C. Kak, "Determination of the identity, position and orientation of the topmost object in a pile", Proc. 3rd Workshop on Comp. Vision, Representation and Control, pp. 38-48, Oct. 1985.

[54] H. S. Yang and A. C. Kak, "Determination of the identity, position and orientation of the topmost object in a pile: some further experiments", IEEE Int. Conf. on Robotics and Automation, pp. 293-298, 1986.

[55] M. Yachida and S. Tsuji, "A versatile machine vision system for complex industrial parts", IEEE Trans. Computers, vol. C-26, pp. 882-894, Sept. 1977.