INTEGRATING VISION AND TOUCH FOR OBJECT RECOGNITION TASKS

Peter K. Allen

Department of Computer Science
Columbia University
New York, NY 10027

CUCS-240-86

Abstract

A robotic system for object recognition is described that uses passive stereo vision and active exploratory tactile sensing The complementary nature of these sensing modalities allows the system to discover the underlying three dimensional structure of the objects to be recognized. This structure is embodied in rich, hierarchical, viewpoint independent 3-D models of the objects which include curved surfaces, concavities and holes. The vision processing provides sparse 3-D data about regions of interest that are then actively explored by the tactile sensor which is mounted on the end of a six degree of freedom manipulator. A robust, hierarchical procedure has been developed to integrate the visual and tactile data into accurate three dimensional surface and feature primitives. This integration of vision and touch provides geometric measures of the surfaces and features that are used in a matching phase to find model objects that are consistent with the sensory data. Methods for verification of the hypothesis are presented, including the sensing of visually occluded areas with the tactile sensor. A number of experiments have been performed using real sensors and real, noisy data to demonstrate the utility of these methods and the ability of such a system to recognize objects that would be difficult for a system using vision alone.

Acknowledgement: This work was supported in part by: NSF grant MCS 82–07294, Air Force grant AFOSR 82-NM-299, NIH grant 5-RO1-HL-29985-02, DEC Corp., Lord Corp. and IBM Corp.

1. INTRODUCTION

There is at present much work going on in the area of sensor design for robotics. Range finders, tactile sensors, force/torque sensors, and other sensors are actively being developed. The challenge to the robotic system builder is to incorporate these sensors into a system and to make use of the data provided by them. Much of the sensor related work in robotics has tried to use a single sensor to determine the structure of objects in an environment[1, 4, 8, 10, 13, 20, 19, 25, 27]. This strategy seems unduly restrictive given the availability of multiple sensing devices. For robotic tasks such as object recognition, in which shape determination of 3-D objects is required, multiple sensors can be used in a complementary fashion to extract more information, in a more reliable way, than a single sensor (e.g. machine vision) strategy [26, 18]. If vision sensing can be supplemented with other sensing information that directly measures shape, more robust and error free descriptions of object structure can result [2].

There are many important issues involved in sensor integration for robotics. Among these are establishing a framework to include new and different sensors; establishing communication and control pathways between the various sensor subsystems; methods for dealing with noise, error and conflict in sensory data; and planning strategies for intelligent use of the sensors. This paper is an examination of these issues within the context of integrating vision and tactile sensing for the task of object recognition. Vision sensing was chosen because of its great promise as a robotic sensor and its use by humans in recognition tasks. Tactile sensing was chosen because it is a low cost robotic sensor that can directly sense the properties of objects we desire, their position and orientation, without regard to visual occlusion. It is a necessary component of any manipulation or assembly system and this paper motivates touch as a natural companion of vision for object recognition.

The paradigm used in this work is model based object recognition in which one of a particular set of known object models is chosen based upon sensory feedback. Figure 1 is an overview of the of the system. The system is divided into 6 main modules: Vision sensing, tactile sensing, sensor integration, matching, verification, and the model data base. The control flow in the recognition cycle is as follows:

1. The vision system images the scene and analyzes all identifiable regions of interest.
2. The tactile system explores each region identified from vision.
3. The results of the tactile and visual sensing are integrated into surface and feature descriptions.
4. The surface and feature descriptions are matched against the model data base, trying to invoke a model consistent with the sensory information.
5. The invoked model is verified by further sensing to see if it is correct.

The experimental hardware is shown in figure 2. The objects to be recognized are rigidly fixed to a worktable and imaged by a pair of CCD cameras. The tactile sensor is mounted on a 6 degree of freedom PUMA 560 manipulator that receives feedback from the tactile sensor and is further controlled by a host processor. The experimental object domain is common kitchen items; mugs, plates, bowls, pitchers, and utensils. The objects are planar as well as volumetric, contain holes and have concave and convex surfaces. These are fairly complex objects which test the modeling and recognition abilities of most existing systems. The objects are homogeneous in color, with no discernible textures. The lack of surface detail on these objects poses serious problems for many visual recognition systems, since there is a lack of potential features that can be used for matching and depth analysis.

The remainder of this paper is organized as follows: Sections 2-7 describe the system's modules in detail and section 8 reports experimental results from sensing and recognizing a number of real objects from the kitchen domain.

2. MODEL DATA BASE

The model data base encodes the high level knowledge about the objects which is needed for recognition. The global structure of the objects which is encoded in the models is used to understand and place in context the low level sensing information. Objects are modeled as collections of surfaces, features and relations, organized into four distinct hierarchic levels. A hierarchic model allows us to do matching on many different levels, providing support or inhibition for a match from lower and higher levels. The models are viewpoint independent and contain relational information that further constrains matches between sensed and model objects. Figure 3 shows the hierarchical model structure for a coffee mug, outlining the decomposition and structure of the models.

The top level of the hierarchy is composed of a list of all object nodes in the data base. An object node corresponds to an instance of a single rigid object. Associated with this node is a list of all the components (subparts) and features (holes, cavities) of this object which make up the next level of the hierarchy. For gross shape classification, a bounding box volumetric description of the object is included. A complexity attribute is also included for each object. This is a measure of the number of features and components that comprise an object and it is used by the matching rules to distinguish competing matches.

2.1. COMPONENTS

The component nodes are the result of a functional *and* geometric decomposition of an object. The components of a coffee mug are the body of the mug, the bottom of the mug, and the handle. A teapot consists of a body, bottom, spout, handle and lid. Each component has an attribute list consisting of its bounding box, surface area, and priority. The priority field is an aid for recognition in which the components are ordered as to their likelihood of being sensed. High priorities are assigned large components or isolated components in space that protrude (handles, spouts). The protruding parts may show up as outliers from the vision analysis. Obscured components, such as a coffee mug bottom, when in a normal pose, are assigned lower priorities. If the object is in a regular pose, then certain parts of the object are more prominent which can aid the matching process. Each component node contains a list of one or more surfaces that make up this functional component and that constitute the next level of the hierarchy.

The subdivision of an object by function as well as geometry is important. In some sense what determines a coffee mug is that it holds a hot liquid as well as having some familiar geometric shape. While no explicit attempt has been made here to exploit the semantic structure of objects, the model maintains a node level in the hierarchy should this be attempted.

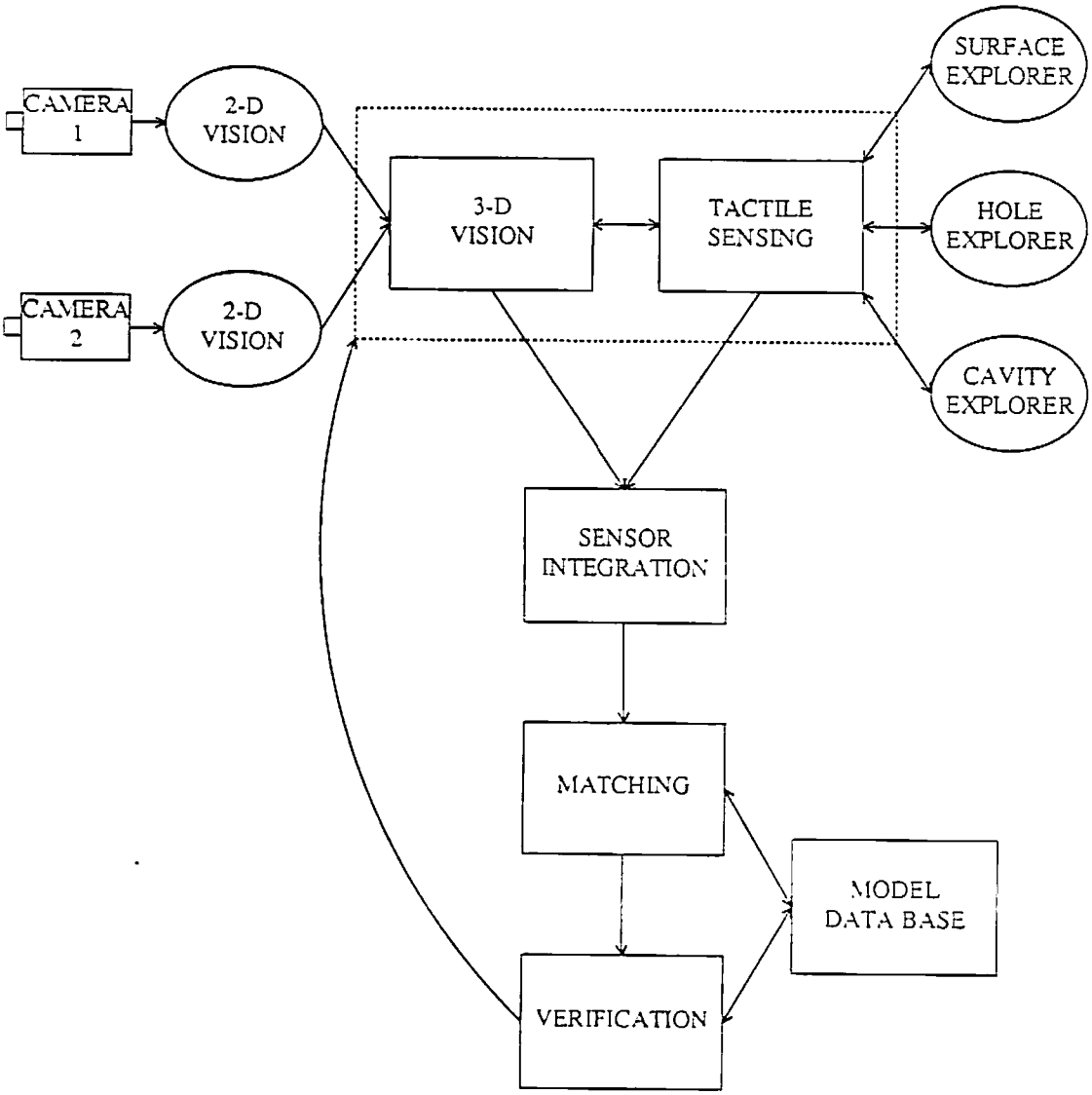


Figure 1: System Overview.

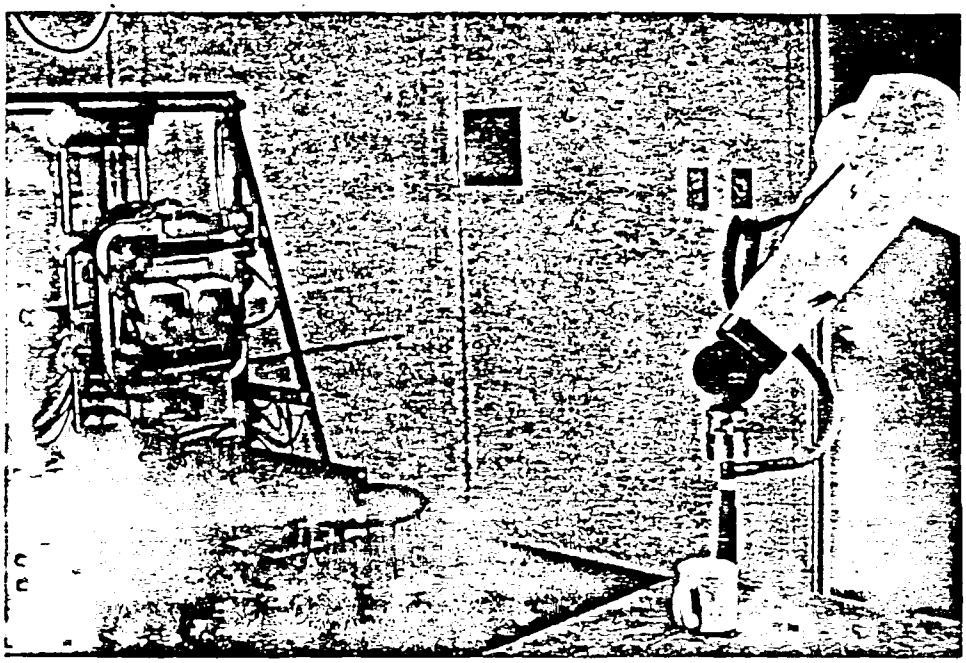


Figure 2: Experimental Hardware.

2.2. FEATURES

Rock [23] has shown that features are important in recognition tasks for humans. The features modeled in the database are holes and cavities. Holes are modeled as right cylinders with constant arbitrary cross section occupying a negative volume. Holes can be thought of as having an approach axis which is perpendicular to the hole's planar cross section. Modeling holes as a negative volumetric entity has implications in matching. Volumetric elements have an object centered coordinate system that contains an invariant set of orthogonal axes (inertial axes). If the sensors can discover these axes, a transformation between model and world coordinates is defined which is a requirement of viewpoint independent matching.

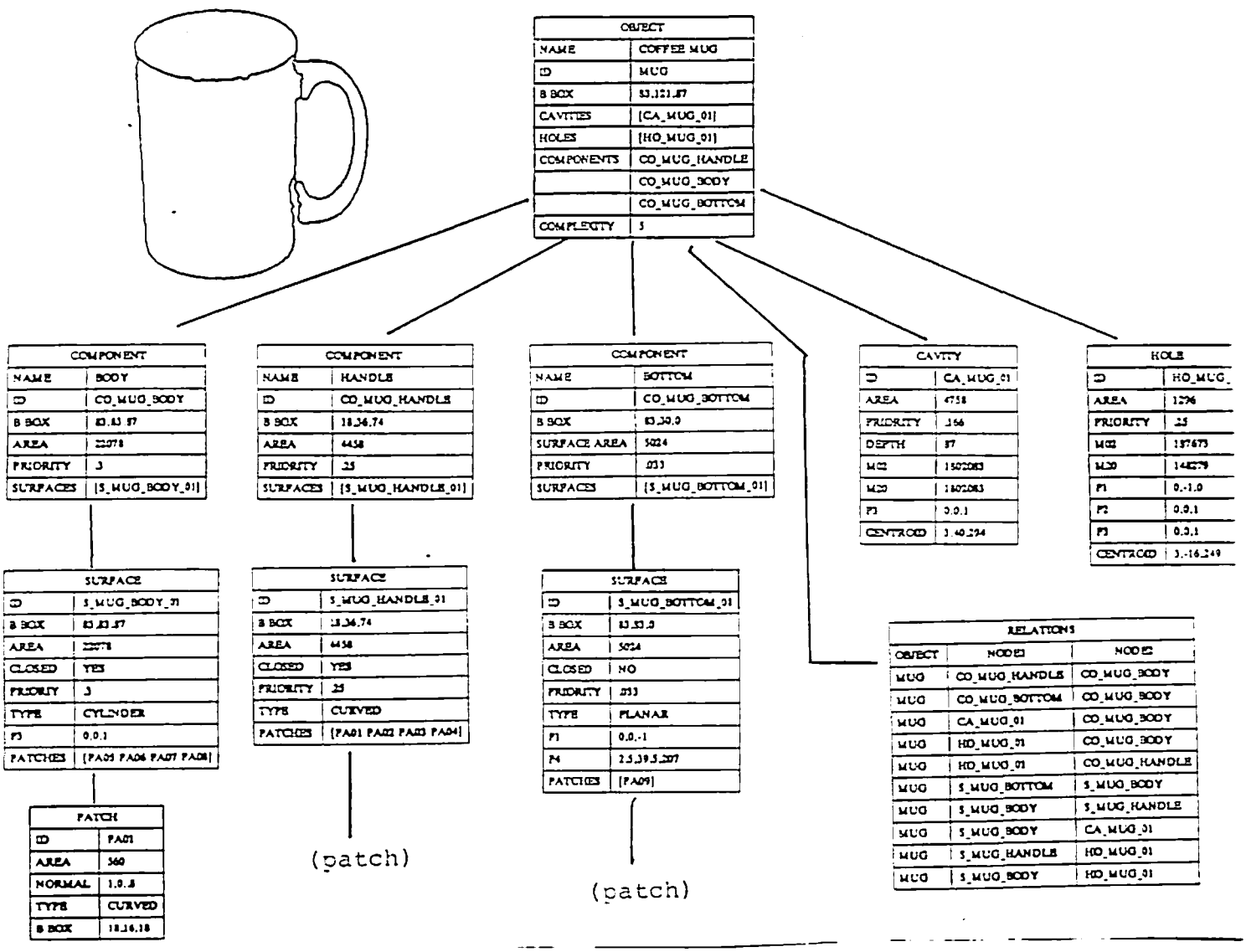


Figure 3: Hierarchical Object Model.

Each hole node contains a coordinate frame that defines the hole. This frame contains a set of orthogonal axes which are the basis vectors for the frame. The hole coordinate frame is defined by the homogeneous matrix H:

$$H = \begin{bmatrix} P_{1x} & P_{2x} & P_{3x} & C_x \\ P_{1y} & P_{2y} & P_{3y} & C_y \\ P_{1z} & P_{2z} & P_{3z} & C_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

P_1 is the axis of maximum inertia of the hole's planar cross section.

P_2 is the axis of minimum inertia of the hole's planar cross section.

P_3 is the normal to the hole's planar cross section.

C is the centroid of the hole's planar cross section.

Besides the coordinate frame, each feature has a set of moments of order 2 that are derived from the planar cross section of the feature's opening.

Cavities are features that are similar to holes but may only be entered from one direction while holes can be entered from either end along their axis. An example is the cavity in the coffee mug where the liquid is poured. Cavities have the additional attribute of depth, which is the distance along the cavity's approach axis from the cavity's opening to the surface below.

2.3. SURFACE LEVEL

The surface level consists of surface nodes that embody the constituent surfaces of a component of the object. The object's components are decomposed by continuity constraints into a number of smooth, continuous surfaces. Each surface contains as attributes a list of bicubic patches that further subdivide it, bounding box, surface area, a flag indicating whether the surface is closed or not and a symbolic description of the surface as either planar, cylindrical or curved. For planar surfaces, a partial coordinate frame is described which consists of the centroid of the plane and the plane's outward facing unit normal vector. For a cylinder, the partial frame consists of the cylinder's axis.

2.4. PATCH LEVEL

The particular form of surface patch that is being used in this research is a bicubic patch known as a Coons' patch [7]. A Coons' patch P is a parametric surface that can be defined as

$$P(u,v) = \sum_{i=0}^{3} \sum_{j=0}^{3} A_i(u)\, B_j(v)\, Q_{ij}$$

where A_i and B_j are the blending functions of the patch and Q_{ij} are coefficients computed from patch data. These patches have been used extensively in computer graphics, computer aided design systems, and object modeling [29, 22]. They possess a number of important features which make them desirable as a 3-D primitive for modeling and for synthesizing surfaces from sensory data. They are interpolating patches constructed from sparse sets of data defined on an arbitrary rectangular parametric mesh. They patches can be joined with C^2 continuity, to form axis independent, complex, composite curved surfaces and their analytic representation allows simple and efficient computation of surface patch attributes. The object domain contains many curved surfaces which are difficult or impossible to accurately model using polygonal networks or quadric surfaces.

Each surface is represented by a grid of bicubic spline patches. Each patch contains its parametric description as well as an attribute list for the patch. Patch attributes include surface area, mean normal vector [22], symbolic form (planar, cylindrical, curved) and bounding box. Patches constitute the lowest local matching level in the system.

2.5. RELATIONAL CONSTRAINTS

One of the more powerful approaches to recognition is the ability to model relationships between object components and to successfully sense them. Relational consistency enforces a firm criteria that allows incorrect matches to be rejected. This is especially true when the relational criteria is based on three dimensional entities which exist in the physical scene as opposed to two dimensional projective relationships which vary with viewpoint.

Each component contains a list of adjacent components, where adjacency is simple physical adjacency between components. The features (holes and cavities) also contain a list of

the components that comprise their cross sectional boundary curves. Thus, a surface sensed near a hole will be related to it from low level sensing, and in a search for model consistency, this relationship should also hold in the model.

At the surface level each surface contains a list of physically adjacent surfaces that can be used to constrain surface matching. These relations are all built by hand, as the geometric modeling system being used has no way of computing or understanding this relationship. The patch relations are implicit in the structure of the composite surface patch decomposition being used. Each patch's neighbors are directly available from an inspection of the composite surface's defining knot grid.

The models have been created by a combination of hand and computer modeling techniques. Figure 4 shows the surfaces that were generated from modeling a plate, a pitcher and a coffee mug. The plate consists of one surface containing 25 patches. The pitcher is made from 24 patches on the handle and 18 on the body. The mug has 4 patches on the body and 24 on the handle.

3. VISION SENSING

The vision processing described here is an attempt to take what is useful and reliable from machine vision and to supplement it with active, exploratory tactile sensing. There is no attempt to try to understand the full structure of an object from vision alone, but to use low and medium level vision processing to guide further tactile exploration, thereby invoking consistent hypotheses about the object to be recognized. The vision processing consists of two distinct phases. The first phase is a series of two dimensional vision routines that are performed on each of the camera images. The second phase is a stereo matching process that yields sparse depth measurements about the object. The output of these modules is combined with active exploratory tactile sensing to produce hypothesis about objects.

Static images of a single object placed on a homogeneous black background are acquired from two CCD cameras which are calibrated with the robotic workspace. The lighting consists of the overhead fluorescent room lights and a quartz photographic lamp to provide enough illumination for the CCD elements. The Marr-Hildreth edge operator [16] is applied to

each of the images and zero-crossings of the convolved images are found. These zero-crossings define homogeneous regions in the image from which region contours are extracted.

The matching phase uses the region contours as input. Isolated zero-crossings not on a contour are discarded, leaving sparse but stable contour match pixels. The matcher then attempts to match contour pixels using the constraints of scan line coherence and zero-crossing orientation and sign. The candidate match pixels are then correlated with regions of small window size centered on each candidate. Only those matches fulfilling the criteria above *and* having a correlation confidence level above 95% are accepted as match points. The outcome of this matching phase is a sparse set of match points on the contours of regions isolated from vision.

There are limitations to the amount and accuracy of the data provided by the vision system. Stereo matching suffers from three main problems. The first is the inability of stereo to handle many candidate match points, such as is found in regularly textured objects. By using only sparse contour data the matcher becomes more accurate with few if any false matches. The second is the error due to quantization on a discrete pixel grid. For the camera geometry used here this can be 4 mm. The location of zero-crossings to subpixels reduces this error to 2 mm. The last problem is the inability of stereo to match horizontally oriented zero-crossings. There is no basis for distinction between locally horizontal matches in a small region. Zero-crossings whose orientation is more than 60° from vertical yield incorrect match results, and are not used by the matcher.

The outcome of stereo matching for a set of digital images of a coffee mug is shown in figure 15. There is sparse 3-D depth data on the contours, containing no horizontal matches. This is clearly *not* enough data to try to recreate surfaces and understand the object's structure. However, the data is accurate and reliable because it has been thinned and abstracted. It allows us to proceed to the next level of sensing with confidence, having sparse but accurate regions identified that can be used for further sensing. Attempts to drive the vision modules beyond this capability will invariably lead to a potentially serious error. The key idea is that *less is more* in the case of multiple sensing. We do not have to rely on this single modality for all our sensory inputs, only those it can *reliably* produce. The matches provided

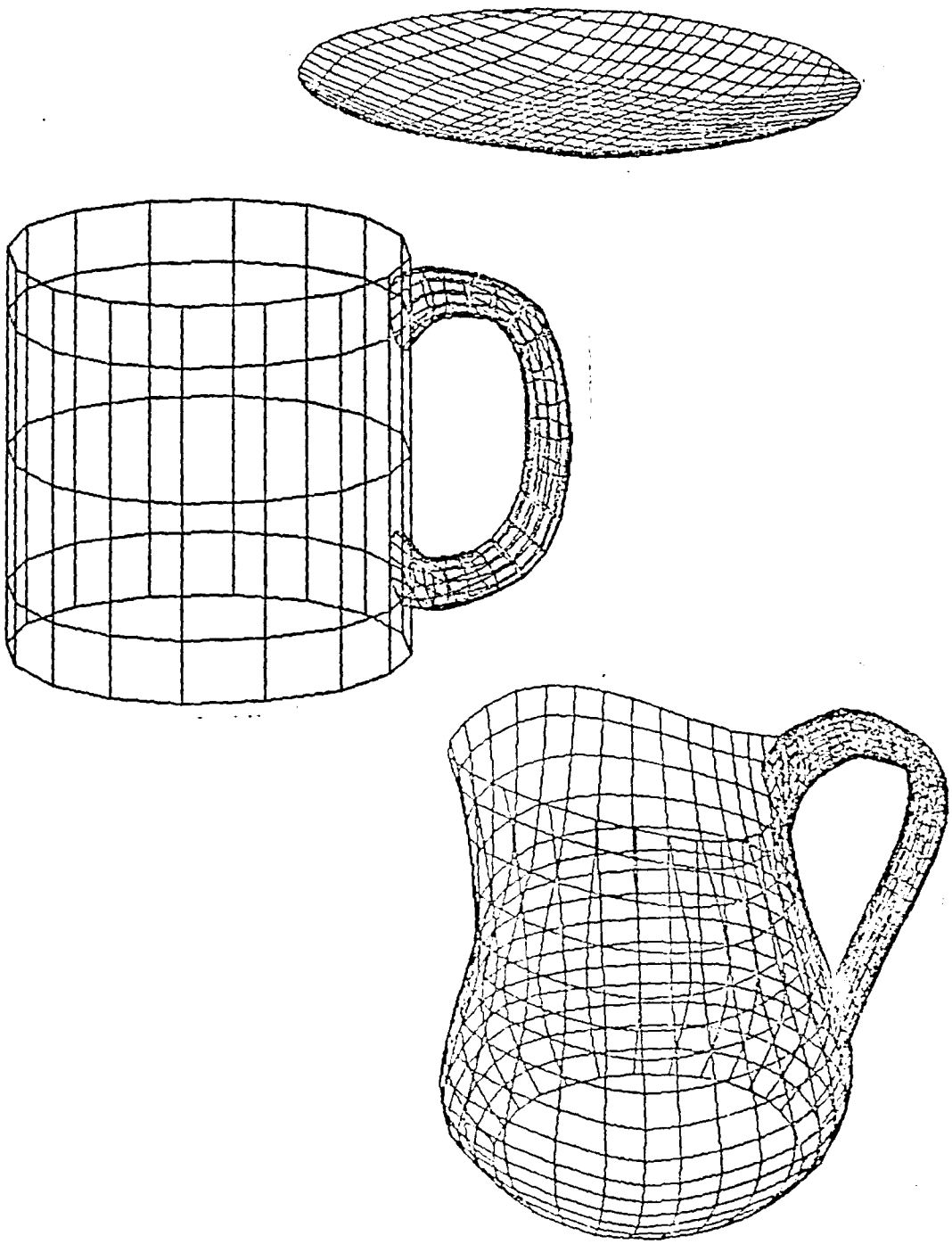


Figure 4: Modeled Surfaces.

by the stereo algorithms are reliable because they are based on contour tokens as opposed to pixels. High confidence levels are established for the matches in order to reduce error. The sparse and conservative matches produced are sufficient to allow tactile sensing to further explore the regions in space.

4. TACTILE SENSING

Tactile sensing is a relatively new and underutilized sensing modality [11]. Previous work in tactile sensing for recognition tasks has emphasised traditional pattern recognition paradigms on arrays of sensor data, similar to early machine vision work [13, 20, 21, 15]. Most sensing has been static in that the sensor is larger than the object and a single touch or "handprint" is used for recognition. Very little has been done on dynamic sensing and integrating multiple touch frames into a single view of an object.

Touch is different from vision in that is an active, exploratory sensing modality. Active touch sensing provides accurate and robust shape information but it extracts its price for this information by demanding powerful control of the medium that makes it difficult to use. Blind groping on a surface with a tactile sensor is a poor and inefficient way of understanding three dimensional structure. Touch needs to be guided to be useful, and the vision data can provide guidance to an active touch sensor.

The experimental tactile sensor used in this research was developed at L.A.A.S in Toulouse, France (figure 5). It consists of a rigid plastic core covered with 133 conducting surfaces that is roughly the size and shape of a human index finger. The geometry of the sensor is an octagonal cylinder of length 228 mm. and radius 20 mm. On each of the eight sides of the cylinder there are 16 equally spaced conducting surfaces. The tip of the sensor contains one conducting surface, and there are four other sensors located on alternate tapered sides leading to the tip. The conducting surfaces are covered by a conductive elastomeric foam. The sensor is connected to a A/D converter that outputs the readings on all sensors in an eight bit gray value and the entire array of sensors may be read in a few milliseconds.

The organization of tactile sensing is on three distinct hardware and software levels (figure 6). The highest level consists of programs on a VAX host that provide high level

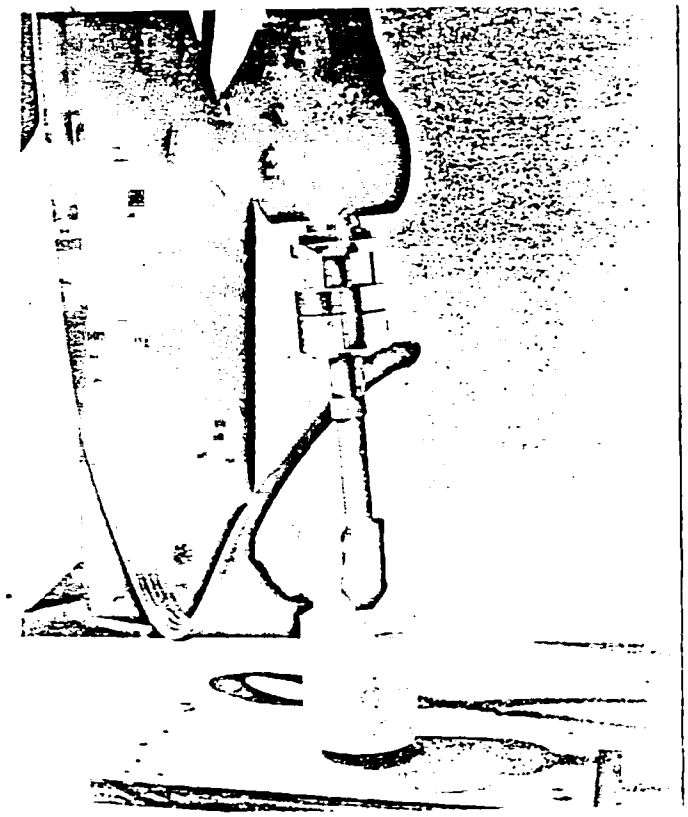


Figure 5: Tactile Sensor.

control information about the regions in space that are to be explored with the sensor. Algorithms have been developed to explore the regions isolated from the vision processing and determine if they are surfaces, holes or cavities. Once a region is identified by tactile sensing, it can be further explored by tactile surface following algorithms that report contact points on surfaces and boundary contours of holes and cavities to the controlling host process. These contacts can be integrated with the 3-D contours from vision to build robust surface and feature descriptions. The intermediate level consists of programs written in VAL-II [28] that run on the PUMA and move the robotic arm based upon feedback from the tactile sensor. The intermediate level receives region exploration parameters via the VAL-II's host control mechanism which then allows it invoke a surface exploration, hole exploration or cavity exploration procedure. These procedures use the feedback from the tactile sensor contacts to control arm motion along the exploration path determined by the high level host control. The intermediate level communicates with the low level sensor system via commands that set thresholds for contacts, requests contact interrupts and requests gray level outputs from

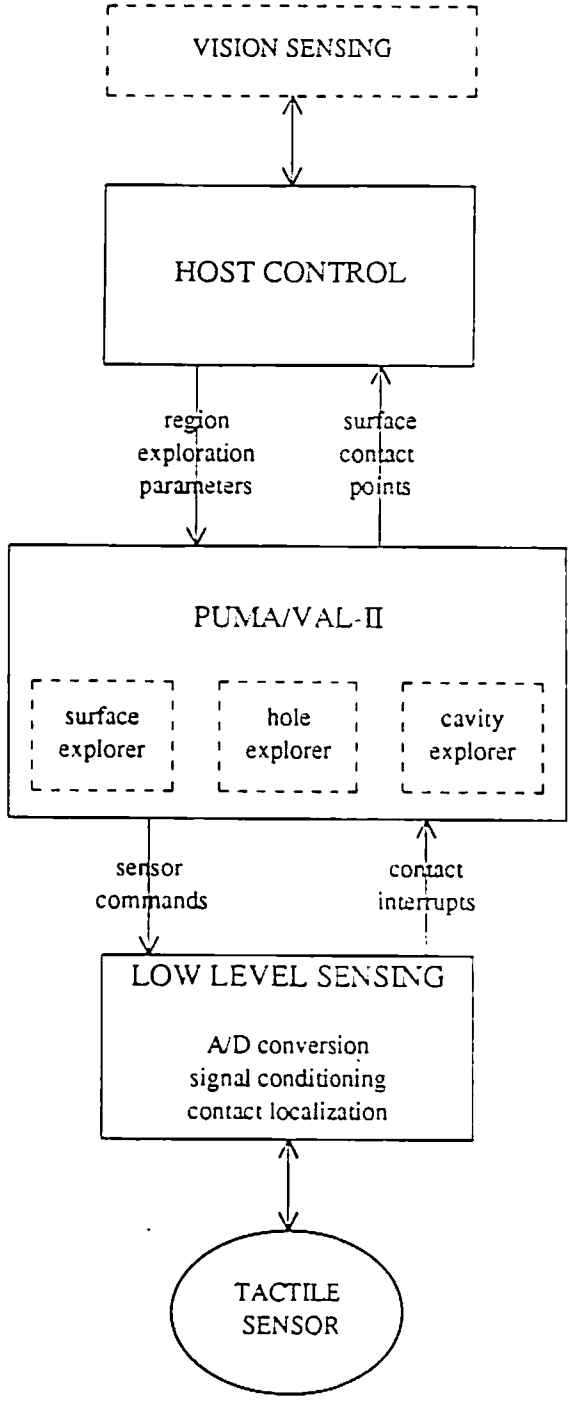


Figure 6: Tactile Sensing System.

arbitrary subsets of the sensor's elements. The low level system is implemented on a microprocessor that samples, digitizes, conditions and localizes the data coming from the tactile sensor, interrupting the intermediate level if a contact of a certain nature occurs.

The classification of a region isolated by the vision system into surface hole or cavity is performed by an intermediate level tactile exploration program. This program controls the motion of the robotic arm and wrist mounted tactile sensor as it explores the region. The program needs as input an approach vector towards the region which establishes the sensor's orientation. The vector is computed by calculating the least square plane P_{lsq} with unit normal $\mathbf{N}_{lsq}$ from the matched 3D stereo points that form the contour of the region. $\mathbf{N}_{lsq}$ then becomes the approach vector for the sensor. The arm control routines will orient the arm so that the tactile sensor's long axis is aligned with $\mathbf{N}_{lsq}$, pointing in the direction of the region's centroid as determined from the vision processing.

The arm is then moved along the sensor's long axis until contact with a surface or it moves beyond plane P_{lsq}, implying the presence of a hole or a cavity. If the sensor is able to travel its full length beyond P_{lsq} without contact, then a hole has been found. If it travels beyond a specified cavity threshold T_{cav} before contact, then it is a cavity. If the region is a surface, a surface exploration program will trace the surface. If it is a hole or cavity, a boundary curve will be traced. The output of these exploration programs will be integrated with the 3-D vision data to form surface and feature primitives (described below) that are used in the matching phase.

5. SENSOR INTEGRATION

Once the tactile system has classified as visually detected region as a surface or a feature, integration procedures are invoked to further sense and quantify the region, allowing the formation of 3-D primitives that can be used by the matching phase.

5.1. BUILDING SURFACE DESCRIPTIONS

The integration of vision and touch data for a sensed surface is done by building a Coons' patch description of the surface. The sparse 3-D contours from vision form the initial patch grid and the description is refined by tactile sensing in the interior of the region. Level 0 surfaces are surfaces comprised of a single surface patch. The information needed to compute a level 0 surface is a 2 x 2 rectangular knot set consisting of points on the surface boundary, the tangents in each of the parametric directions at the knots and the twist vectors (cross derivatives) at the knots (figure 7). The knot points should be chosen at points of high curvature on the boundary curve and need to be spaced uniformly in each of the parametric directions. The algorithm for choosing points of high curvature on a contour is a modification of an algorithm originally proposed by Johnston and Rosenfeld [24]. This algorithm analyzes the boundary contour's curvature at different scales, choosing local maxima along the contour. The knot points are then chosen by maximum curvature and distance along the boundary contour in order to preserve equal parametric length for opposite boundary curves.

The tangent vectors in each of the parametric directions must also be calculated. The contour of the region contains a series of three dimensional data points obtained from stereo matching that define four boundary curves on the surface. These curves are approximated by a least square cubic polynomial parametrized by arc length which is then differentiated and scaled to yield tangent vector values at the knots.

The twist vectors are more difficult to estimate. If the parametric directions on the surface are along the lines of curvature of the surface, then there is no twist in the surface and the twist vectors are zero. In practice, these vectors can be set to zero with minor effects on the surface. This assumes that the parametrization of the surface has been chosen wisely, with corner knot points chosen at places of high curvature or discontinuity along the boundary and spaced uniformly in both parametric directions.

A level 0 surface is built from vision data only and is not an accurate description of the underlying surface since it lacks information about the interior of the surface. The tangents which are estimated from stereo match points are inaccurate along contours that are horizontal due to the lack of stereo match points. Figure 7 describes the method of building higher level

surfaces. A level 1 surface is formed by adding tactile traces across the single surface patch defined in level 0, and a level 2 surface is formed by adding tactile traces to each of the 4 patches defined by level 1 creating a new surface with 16 patches. This method is hierarchical, allowing surfaces of arbitrary level to be computed. The only restriction is that the new composite surface is globally computed.

The traces begin at the point of surface contact found in the initial exploration of the region found from vision processing. The sensor then traces in the direction of the midpoints of the level 0 boundary curves, using the surface contacts from the tactile sensor to control the robot arm motion.

The movement vector $\mathbf{M}$ along the surface is determined by:

$$\mathbf{M} = \sum_{i=1}^{3} w_i \, \mathbf{G}_i$$

w_i are the weights for each of the vectors $\mathbf{G}_i$.

$\mathbf{G}_1$ is the unit vector in the direction of the boundary curve midpoint.

$\mathbf{G}_2$ is the unit vector formed from the previous two contact points.

$\mathbf{G}_3$ is the unit vector that preserves equal parameterization.

$\mathbf{G}_1$ is needed to make progress towards the boundary edge. We will want to make progress towards the boundary at each movement step. However, with concave and convex surfaces, cycles can occur as the trace progresses. $\mathbf{G}_2$ is used to maintain a path's direction. Once we start moving in a certain direction we do not want to stray too far too fast from that path. This vector is an ''inertia'' vector helping the sensor stay on a steady course. $\mathbf{G}_3$ is needed to keep the parameterization of the surface patches uniform, and this vector moves the trace in the direction to preserve equal parameterization. This vector is the unit resultant of the vectors from the present contact point on the surface to the endpoints of the boundary curve that the trace is approaching.

The points reported during these traces are combined into cubic least square polynomial curves that are differentiated and scaled to calculate the tangential information needed at the boundaries. The boundary curves tangents computed from vision data are updated to include

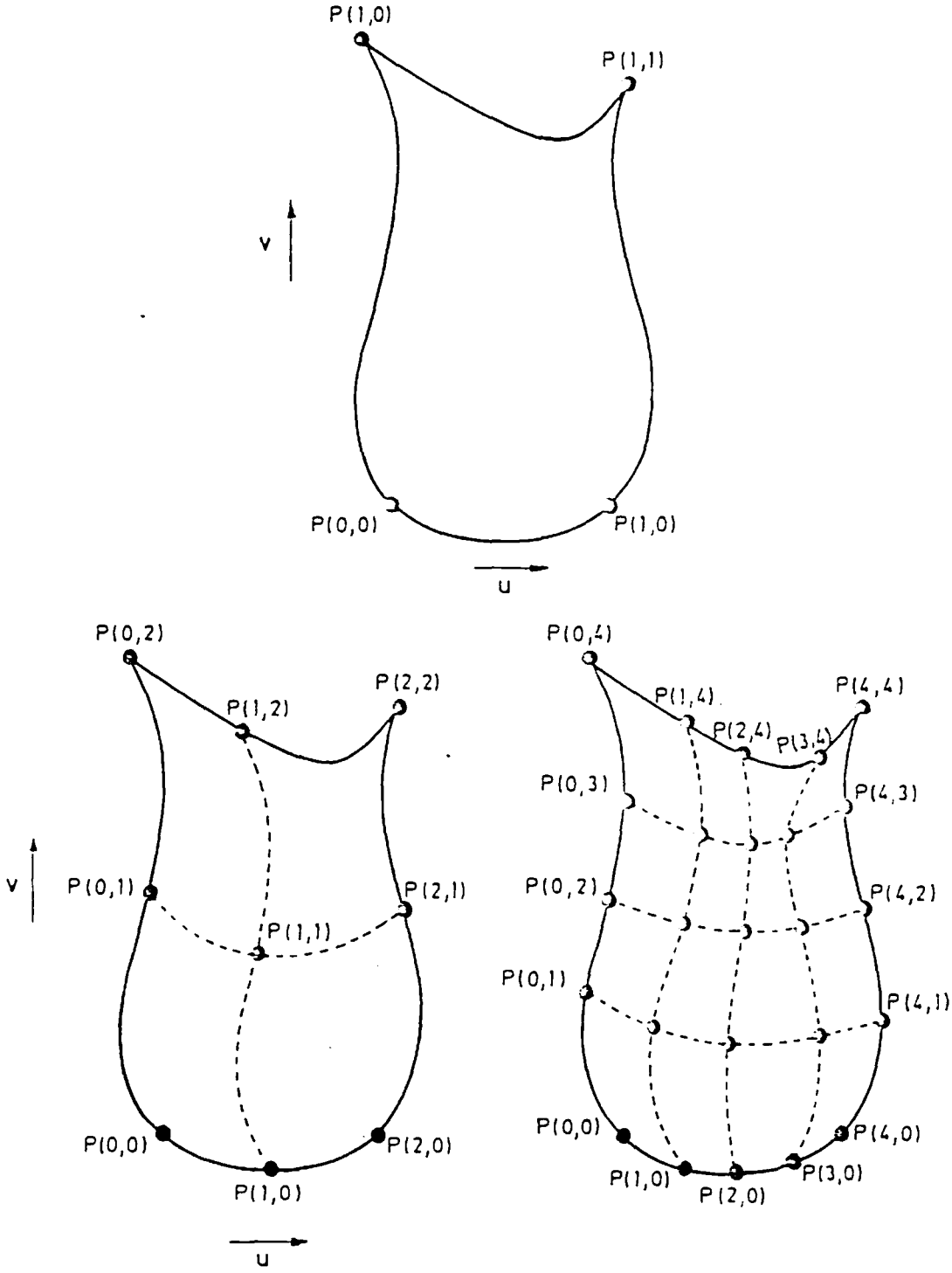


Figure 7: Level 0, 1 and 2 Composite Surfaces.

the new tactile information, which fills in areas that lack horizontal detail from the stereo process.

Figure 8 shows the level one surface that results from active tactile sensing of the front surface region of a pitcher. The surfaces are accurate and built from sparse amounts of data. The analytic nature of these surfaces allows stable and accurate symbolic descriptions based upon the surfaces Gaussian curvature to be computed, classifying these surfaces as planar, cylindrical or curved.

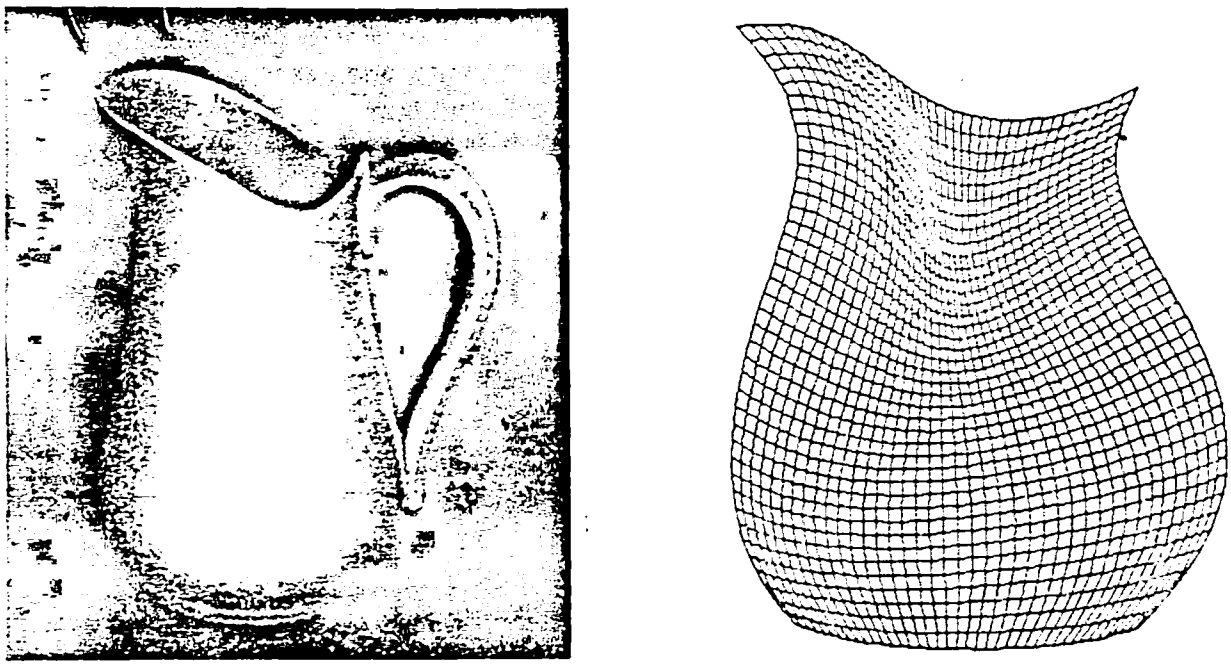


Figure 8: Digital Image and Level 1 Surface of a Pitcher.

It is important to note that the vision processes are supplying the justification for building smooth curvature continuous surfaces from a region. If the region were not a smooth surface, then zero-crossings would have appeared inside the region, precluding the assumption of smoothness. The lack of zero-crossings, or the "no news is good news" criteria established by Grimson [9] supports this method and in fact is the reason it succeeds in interpolating the surfaces well.

5.2. BUILDING FEATURE DESCRIPTIONS

If the region exploring algorithm determines that the region is a hole or cavity, a different tactile tracing routine is used to determine the boundary curve of the feature. The algorithm begins by moving the sensor just beyond the least square plane P_{lsq} of a region's contour points, aligned with N_{lsq}. It then proceeds to move in a direction perpendicular to N_{lsq} until it contacts a surface. Once the surface is contacted, the sensor moves along the bounding surface staying perpendicular to N_{lsq}, recording the contact points until it reaches the starting point of the trace.

This can be a noisy procedure as many of the tactile sensor's contacts become activated in a small tight area such as the hole in the handle of a coffee mug. The spatial resolution of the sensor contacts also contributes to this phenomena. The data is not continuous, but is a set of ordered contact points that need to be smoothed and this is done by approximating the series of linked contour points with a periodic spline curve which matches derivatives at the endpoints. Figure 17 shows the smoothed boundary curve created from sensing the hole in the coffee mug. This boundary can then be used to compute cross sectional area and moments for matching against the model data base of objects.

6. MATCHING

The low level vision and tactile algorithms provide a set of three dimensional surface and feature primitives that are used by the matching routines to determine what the object is and its orientation. The matching routines try to find an object in the model data base that is consistent with the surface and feature information discovered by the sensors. The intent is to invoke a uniquely consistent model from the three dimensional surface and feature primitives discovered. If more than one consistent object is found in the data base, a probabilistic measure is used to order the interpretations. Once a consistent interpretation is found, a verification procedure is begun. This requires the matcher to calculate a transformation from the model coordinate system to the sensed world coordinate system. This transformation is then used to verify the model by reasoning about the slots in the model data base that are not filled. The initial choice of a model is made easier by the three dimensional nature of the

primitives, allowing matching of higher level attributes rather than sets of confusing and noise filled point data. The rules used for invoking a model are such that no a priori choice of features or surfaces is needed; all the structural parts of the model are candidates for matching. The object recognition system has no way of knowing what features or surfaces will be sensed from a particular viewpoint. It must be able to invoke a model based upon any identifiable part of the model [3].

The matching phase is the most difficult of all the modules since it requires the system to do high level reasoning about objects and their structure based upon incomplete and partial sensor information. The approach taken here is to develop a rule based system that will allow experimentation and modification of sensing strategies. Some of the rules and strategies implemented are discussed below and development of new rules and strategies are a focus of our current work.

Model instantiation is done by first pruning the space of object models that are not consistent with two global criteria, physical size and gross shape. All objects in the model data base consistent with these criteria are then further matched according to feature and surface attributes as determined by the integration of vision and touch sensing.

One of the benefits of using active tactile exploration is that physical size constraints can be used for global discrimination. Nevatia and Binford [17] and Brooks [6] have shown the utility of using physical size constraints in recognition tasks. The tactile sensor can be moved into the workspace to trace the global outline of the object to determine its bounding box. This procedure also puts coarse bounds on the location of the object which can be used by the verification procedures later. Gross shape is able to prune based upon number of features discovered and whether surfaces are classified as planar, cylindrical or curved based upon the Gaussian curvature of the sensed surface [12]. Sensed surfaces constrain the set of consistent object models less tightly because the sensors discover patches of possibly larger surfaces (the aperture problem). A curved surface in the model may have cylindrical regions, which may be sensed as a cylindrical partial patch. Therefore, gross shape discrimination must be conservative in matching curved surfaces.

Feature attributes are used as a discrimination tool to invoke a consistent model. The constant cross section of the feature can be used to define a set of moments that can be used to match the cross section with a sensed feature. Moment matching was first described by Hu [14] who described a set of seven moment invariants involving moments of up to third order. At the instantiation level the moment M_{00}, which measures the area of the planar cross section, and the second order moments $M_{02} + M_{20}$ are matched between the sensed and model systems. The latter measure is scaled to reflect the difference in M_{00} when it is matched. In the case of cavities, the depth attribute is also used as a matching criteria.

Surface matching tries to match on two attributes, area and type of surface. The sensor is not capable of sensing accurately parts of the model with fine structure such as the handle of the mug. The area criteria effectively culls out small feature matching and leaves the task of larger shape correspondence. The set of possible consistent interpretations is restricted further by maintaining relational consistency between the sensed regions and the model nodes. The relational constraint used is adjacency. If two sensed regions in space are physically adjacent, then the model nodes that these regions match with must also be adjacent.

7. VERIFICATION

Verification can be viewed as slot filling, where the instantiated model's nodes are either filled, representing a sensed match, or unfilled. Verification then becomes a process of reasoning about unfilled slots. Once a model is instantiated, a transformation between model coordinates and sensed world coordinates must be computed. This transformation will allow the knowledge embedded in the model coordinate frame to be used in the sensed world frame. By transforming model surfaces and features to the sensed world frames, verification of unrecognized slots in the model can proceed since their assumed location is now computable with this transformation. This knowledge enables the sensors to explore regions that were not seen in the initial sensing and to explore visually occluded areas with tactile sensing. The transformation may be computed with feature information or surface information. In some cases, a partial transformation may be computed that will allow further verification sensing.

7.1. MATCHING FEATURE FRAMES

Each feature in the data base is associated with an object centered coordinate frame. Once the models and their frames are developed, mappings from one feature frame to another are readily computable. Figure 9 shows the frames C_m and H_m which are object centered frames defined for a coffee mug's cavity and a hole in the model coordinate system. The relative transform between the hole frame and the cavity frame R_{hcm} can be defined as:

$$C_m = H_m : R_{hcm}$$

$$R_{hcm} = H_m^{-1} : C_m$$

Similarly, the transformation from modeled cavity to modeled hole R_{chm} is:

$$R_{chm} = C_m^{-1} : H_m$$

Because these are relative frames, discovering one of the model frames in the sensed coordinate space will define the other feature in the sensed coordinate space. Assuming we know the match between the hole in sensed world coordinates with frame H_s and the model hole with frame H_m then the cavity in sensed world coordinates is defined by frame C_s:

$$C_s = H_s : R_{hcm}$$

The determination of the new feature frame in sensed world coordinates is important to the verification process. If an unfilled feature slot is seen, then the feature's frame in sensed coordinates is available through the relative frame mapping. The frame for a feature defines the axis of the hole or cavity in sensed world coordinates which is then used as an approach vector to sense the unseen feature even if it is occluded.

Feature frames may be only partially defined as is the case with rotationally symmetric features such as a circular cavity or hole. The approach axis of these features is well defined, but the principal axes of inertia of the cross sectional opening are not. However, the frame matching technique discussed above can still determine within this rotational parameter the new sensed frame. An example of this is given in section 8, where the tactile sensor is able to sense a visually occluded hole.

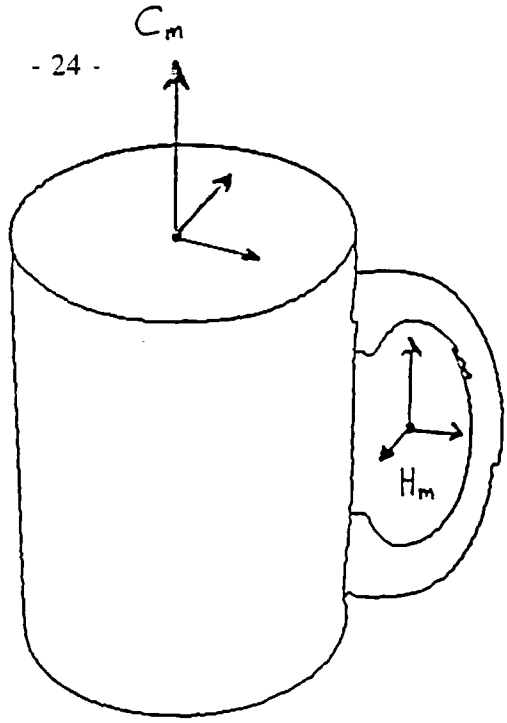


Figure 9: Relative Feature Frames.

7.1.1. MATCHING SURFACE FRAMES

Matching of surfaces is more difficult because a unique surface frame is not as easily sensed as a feature frame. Planar and cylindrical surfaces have one well defined frame vector which is the plane's normal and the cylinder's axis. Curved surfaces in general do not have any such natural embedded frame. In the case of planar and cylindrical surfaces, the one axis which is defined will allow defining the transformation up to a rotational parameter about that axis and a translation. In the case of the plane, the plane's centroid is also computable and this will supply the translational component of the transformation. This can be used in conjunction with other feature and surface matches to constrain the sensed frame.

The analytic nature of the surfaces created from vision and touch allows computation of differential geometry measures such as lines of curvature, principal directions, and Gaussian curvature. Brady, Ponce, Asada and Yuille [5] have suggested that certain lines of curvature that are planar might be significant in terms of recognizing structure. For example, the only planar lines of curvature on an ellipsoid are the lines formed by the intersection of the symmetry planes with the surface. Discovery of lines such as these is feasible with the representation used, and may lead to more robust recognition methods for curved surfaces.

8. EXPERIMENTAL RESULTS

This section details the experiments that were conducted to test out the approaches developed in the previous sections. The experiments show that integrating vision and touch is a viable method for recognition, particularly when compared to standard vision processing. The experiments reported have been run using real objects and real noisy sensors. In addition, the tactile sensor being used is relatively crude in terms of spatial resolution compared to newer devices. Despite these shortcomings, the approaches to matching discussed previously work well in a number of important cases. The implementation of the matcher consists of a set of PROLOG goals that match sensed regions with model nodes. The model data base is implemented as a set of PROLOG facts that are indexed in a hierarchical manner. The data base consists of eight kitchen objects: pitcher, mug, spoon, teapot, plate, bowl, drinking cup, pot. Four of these objects (pitcher, mug, plate, bowl) were used in experiments to test the matcher and its ability to correctly identify the objects. The main intent of these experiments is to show 1) the utility of the methods presented and 2) the ability of touch and vision to succeed in situations that vision alone would find difficult.

The first experiment tried to recognize a planar salad plate. The digital images and stereo matches are shown in figure 10. The images yielded few feature points that could be matched to determine depth as expected with a smooth homogeneous surface. The stereo matcher was only accurate in matching zero-crossings up to 65° from vertical, yielding sparse and incomplete depth information. An image such as this would pose large problems for a vision system alone; the data is too sparse to support a consistent visual hypothesis. The region analysis revealed only a single region to be explored which was the central area of the plate. The tactile system explored the plate and built the surface description shown in figure 11 by integrating the touch and vision data into a level one surface description. The surface was sampled at small intervals in parameter space calculating the Gaussian curvature and confirming its planar nature. Figure 12 shows the computed surface normals on the plate, verifying its planar appearance.

The normal of the least square plane fitted to the surface was the estimate for the orientation of the object. No other orientation parameters were available since the plate was

symmetric about the surface normal. The sensed plate's estimated surface normal was within 6 ° of the actual plate orientation on the table.

8.1. EXPERIMENT 2

The second object imaged was a cereal bowl. The digital images and stereo matches are shown in figure 13. The images are similar to the plate in experiment 1. The only depth cues are monocular, where small shading gradients exist but which elude the zero-crossing edge detector. This is an excellent example of the discriminatory power when tactile sensing is added to vision. The region analysis yields one region to explore with the tactile sensor. Upon exploration, a level one surface of the bowl was computed and is shown in figure 14. The tactile sensor did not find a surface until it had passed 40 mm. beyond the plane of the region's contour determined from vision. This prompted a cavity trace in addition to the surface trace.

The matcher tried to match the surface and the cavity with an object in the database. The combination of the curved surface and cavity (with measured feature moments and depth) was sufficient to invoke the correct model. The estimate of the object's orientation in space was the angular difference between the actual cavity axis and the sensed axis which was approximately 5 °.

The initial visual data for experiments 1 and 2 were almost identical. Only by using touch sensing did the surface's depth become apparent. The discovery of a cavity allowed the system to discriminate between two potential surface matches. The combination of surface and feature information reduces the likelihood of multiple consistent models being found.

8.2. EXPERIMENT 3

The third experiment imaged a coffee mug. In this image the hole, cavity, handle and body of the mug were all visible. The digital images and the stereo matches are shown in figure 15. The region analysis yielded 4 separate regions to explore. The first region explored was the cavity. The second region explored is the mug's main body for which a surface patch was built and is shown in figure 16. This surface patch is a level one patch

built from vision and touch and very closely approximates the cylindrical surface of the mug. The analysis of the patch's Gaussian curvature classified the patch as a cylinder.

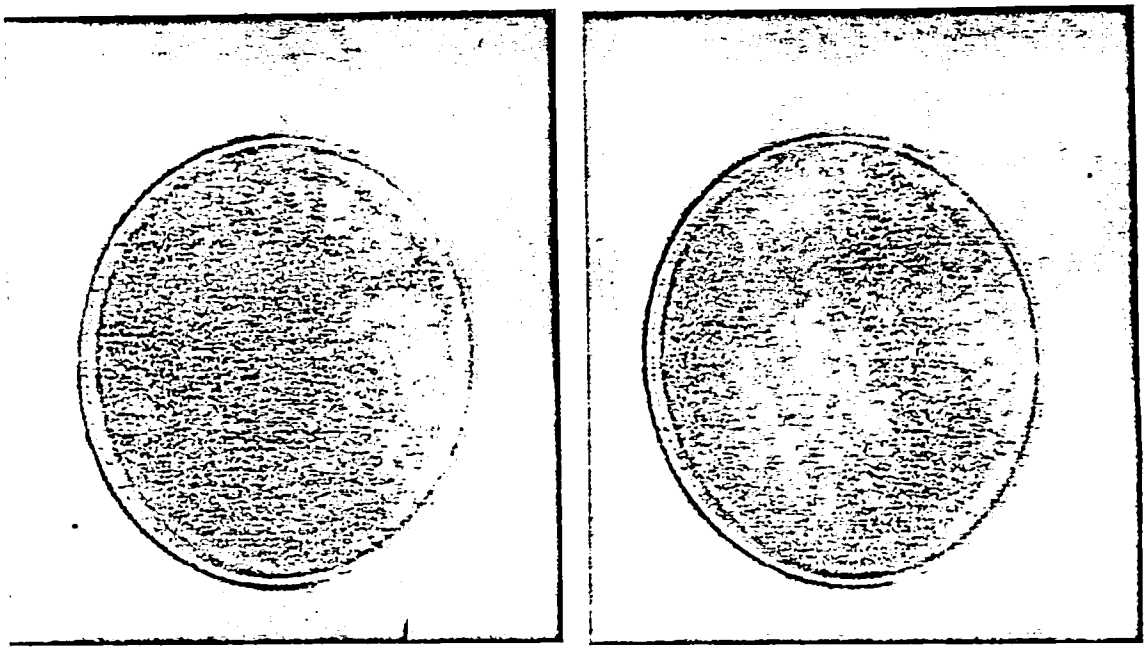
The hole was found after the region exploration algorithm penetrated the region defined from vision processing and did not contact a surface. The hole was traced by the tactile sensor and the smoothed boundary curve shown in figure 17 was computed from the contact points on the holes boundary.

The matcher was presented with an abundance of sensed region information to try to instantiate a model. The cylindrical surface that was computed matched a number of objects in the database (pot, coffee mug, drinking glass) as did the cavity (drinking glass, coffee mug). The hole was not found in the drinking glass (an identical object in the database to the mug but without a hole or a handle) but matched with the coffee mug, yielding a unique choice of object. The cylindrical surface axis and the cavity axis are parallel in the model and the agreement between these two axes and the actual orientation was quite close ($< 5\ °$).

The handle of the mug is too small and fine for the sensor to adequately build a patch description. It can be verified as a surface with the sensor, but attempts at building a patch description failed due to the sensor's much larger size. This experiment shows the many ways an object can be recognized. Holes, cavities and surfaces are all able to be used to both recognize and correctly identify orientation parameters for the objects. This is important in that certain viewing angles may present a confusing region that cannot be sensed accurately. However, if one of the regions is able to be sensed accurately, then a partial match can be established leading to later recognition.

8.3. EXPERIMENT 4

In this experiment, the coffee mug was imaged with the handle occluded. The objects in the data base that will match with these two regions (body surface and cavity) are the drinking glass without a handle and a mug with a handle. From this visual angle there is no way that the two objects can be distinguished. The instantiation module will pick both objects to be verified.



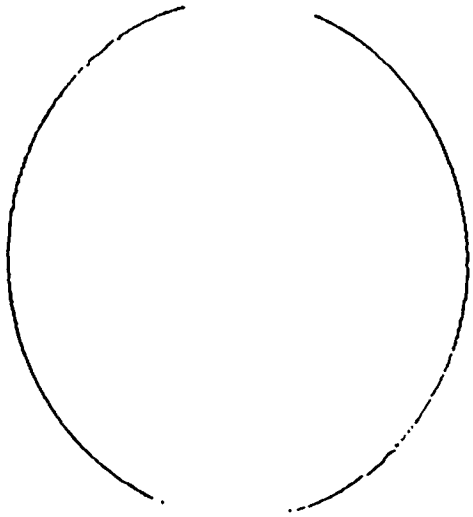


Figure 10: Digital Images and Stereo Matches, Plate.

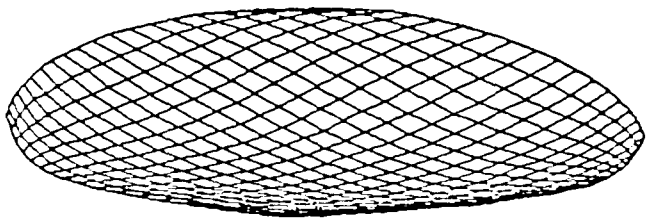
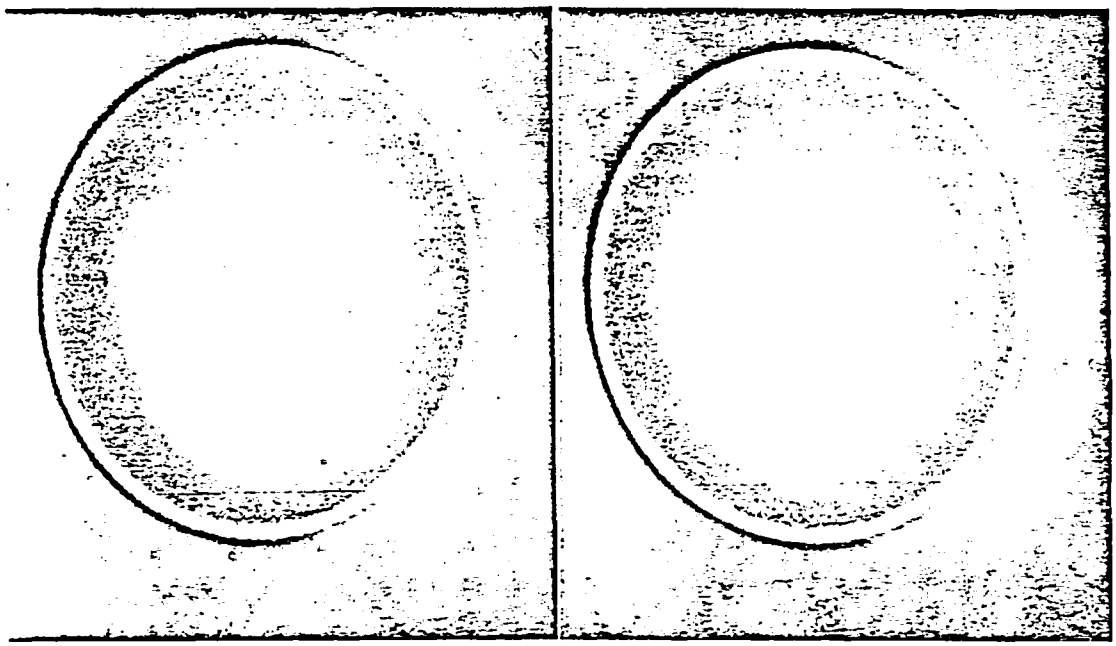


Figure 11: Level 1 Surface, Plate.

Figure 12: Sensed Surface Normals, Plate.

It can be determined that the object is a mug by verifying the occluded hole. If it is a mug, the hole lies in the occluded area which is shown in figure 18. The bounds on this volume are known from the vision and touch sensing that has already been performed. The handle can be located by knowing the relative feature frames between the sensed cavity and the hole. The cavity, however, does not possess a unique frame; it is rotationally symmetric, leaving a degree of freedom in its internal frame which is the rotation about its approach axis.



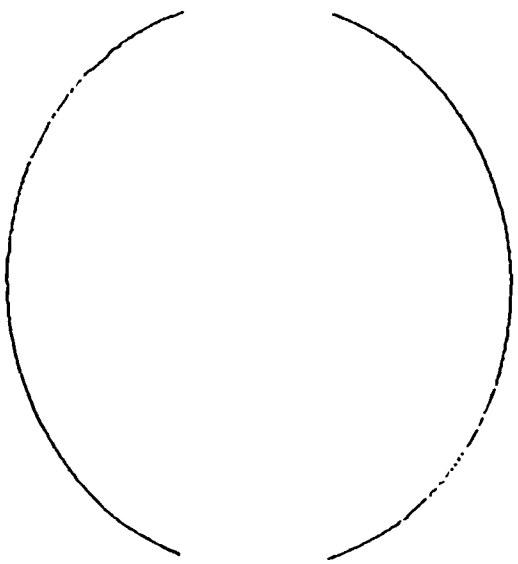


Figure 13: Digital Images and Stereo Matches, Bowl.

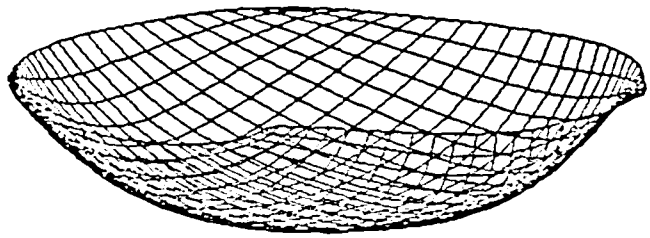
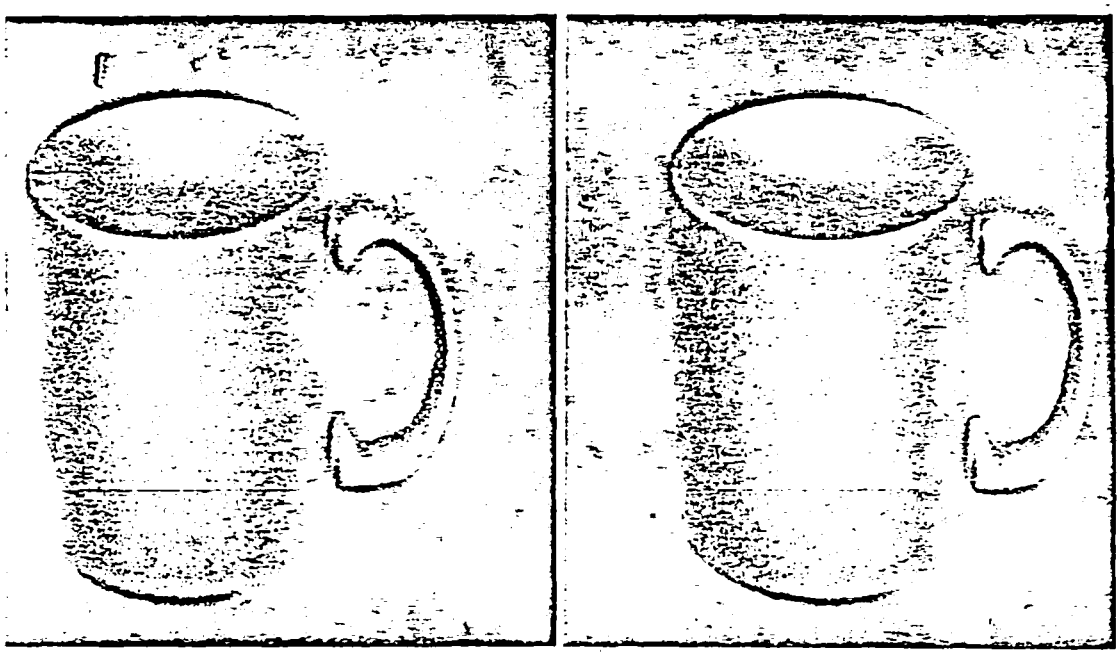


Figure 14: Level 1 Surface, Bowl.

This degree of freedom can be exploited to reason about the location of the hole. The fixing of the cavity's approach axis in space means that the hole centroid is confined to lie in a circle centered at the cavity and swept out about the cavity's axis. Computing this circle gives a set of three dimensional points which represent possible locations of the hole's centroid. Intersecting this circle with the known occluded volume yields a possible set of locations of the hole. Each of these locations is associated with a particular fixing of the rotationally symmetric axes about the cavity's axis. The approach is to fix the cavity's rotationally symmetric axes at an angle of rotation that is midway between the angles that bring the hole into occlusion and bring it out. Once this is defined, it yields an approach axis for the hole which the tactile sensor can then use to probe the hole. In the experiment, the hole was found this way, rejecting the drinking glass match and accepting the mug match. Figure 19 shows the sensor searching for and finding the hole in the visually occluded area.

This last experiment shows the power of this approach to object recognition. Multiple sensors were used synergistically to invoke a possible set of objects. High level reasoning about the object's structure that is encoded in three dimensional models allowed further verification sensing to successfully discriminate between the objects. The knowledge about the three dimensional world (the occluded volume) and the object's geometry (which is encoded in the model) can be used to perform active sensing in occluded areas.



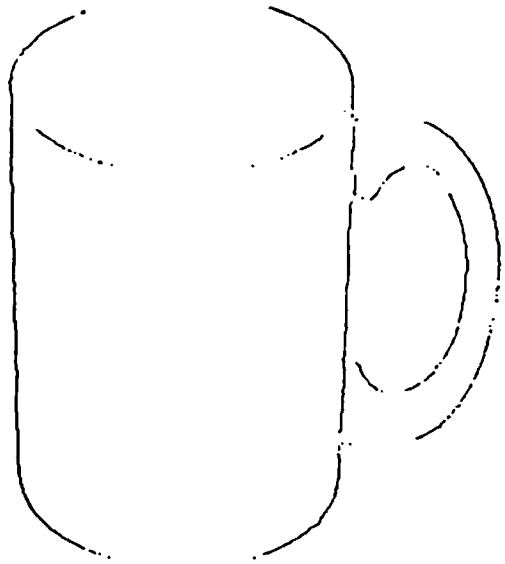


Figure 15: Digital Images and Stereo Matches, Coffee Mug.

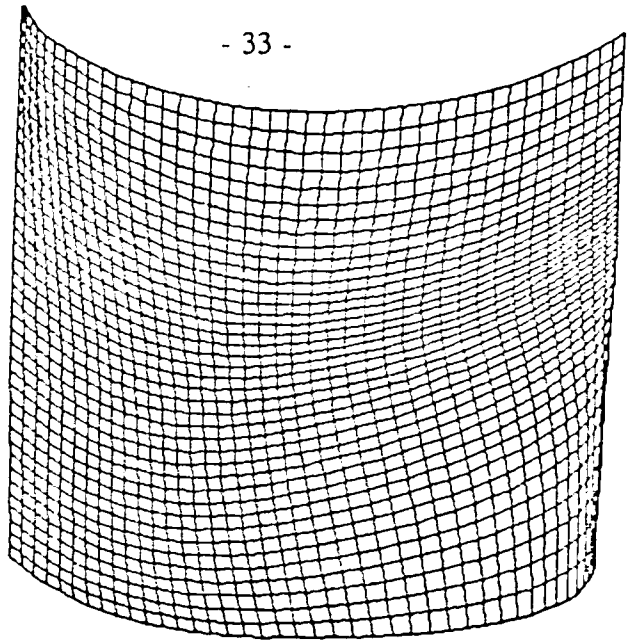


Figure 16: Level 1 Surface, Coffee Mug.

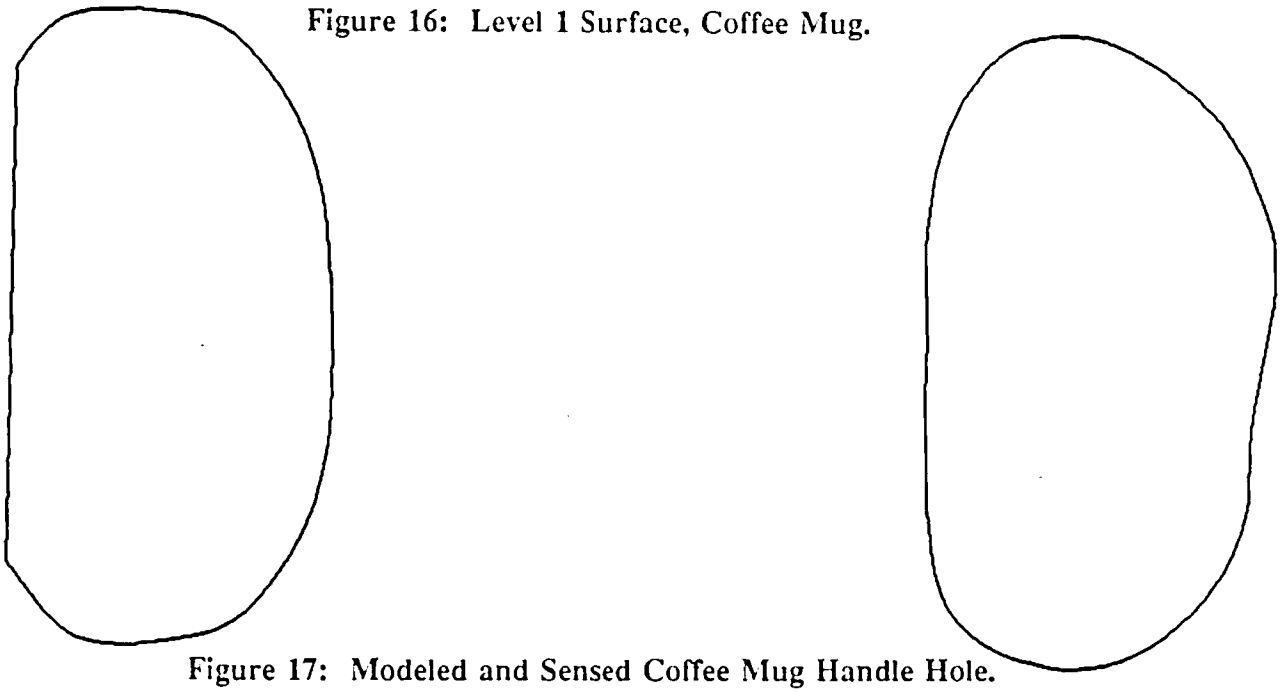


Figure 17: Modeled and Sensed Coffee Mug Handle Hole.

9. SUMMARY

This research has attempted to improve robotic performance in a real noisy object domain by integrating multiple sensors. The use of multiple sensors has provided more robust and accurate sensory data that can be combined into three dimensional primitives that facilitate matching and an understanding of the underlying structure of the objects. The ability to sense actively demands higher levels of control than with passive sensors, including the ability to reason at a high level about object structure. This reasoning capability needs to be further

developed and is a natural extension of this work. allowing tasks beyond recognition to be attempted in a multi-sensor environment.

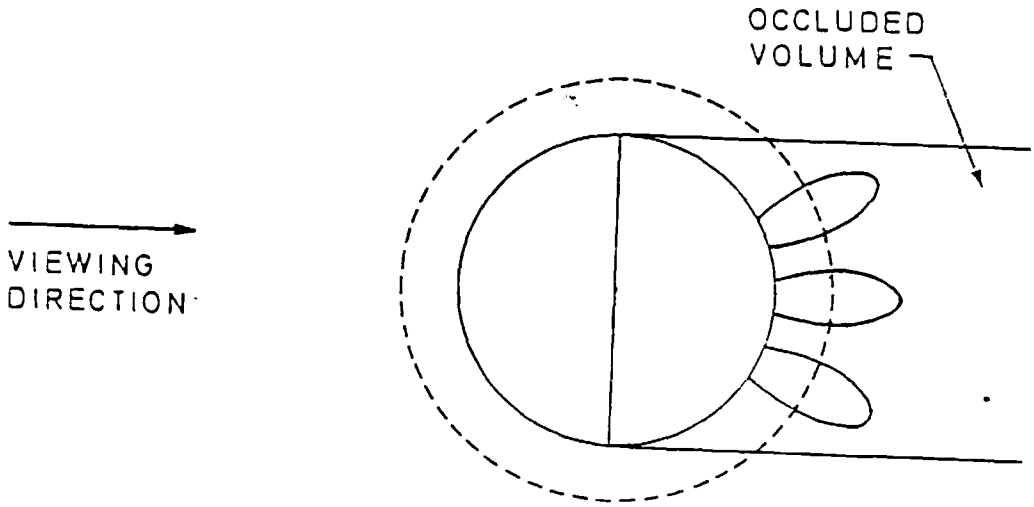


Figure 18: Occluded Volume of Coffee Mug.

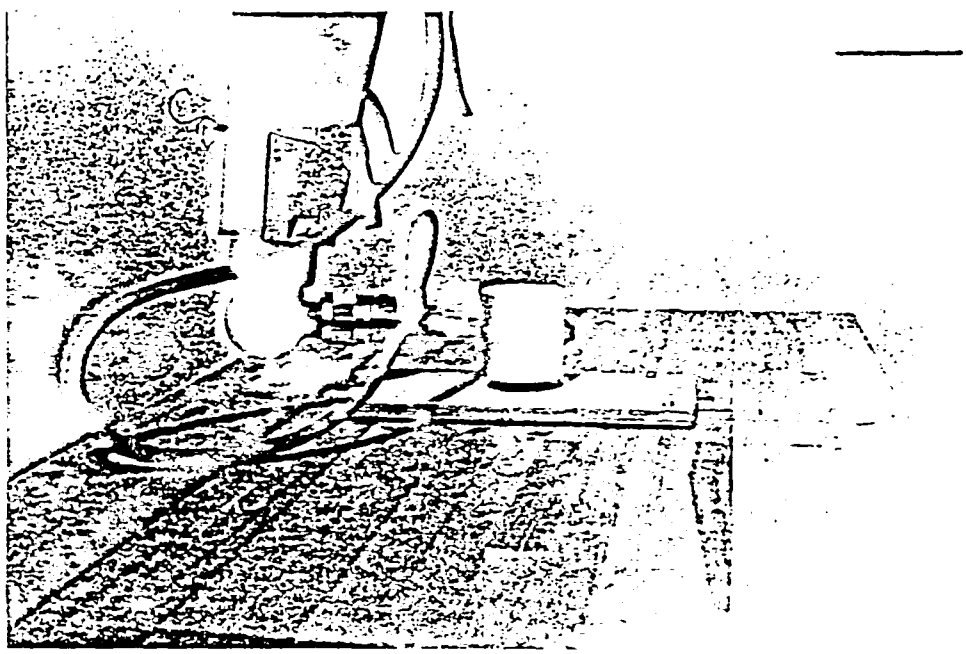


Figure 19: Tactile Trace of Occluded Hole.

References

1. Agin, G., "Representation and description of curved objects," Stanford University A.I. Memo, October 1972.

2. Allen, Peter, "Object recognition using vision and touch," Ph.D. Dissertation, University of Pennsylvania, September 1985.

3. Allen, Peter and Ruzena Bajcsy, "Converging disparate sensory data," *Proc. 2nd International Symposium on Robotics Research*, Kyoto, August 1984.

4. Bolles, R. C., P. Horaud, and M. J. Hannah, "3DPO: A three dimensional part orientation system," *Proc. 8th IJCAI*, Karlsruhe, Germany, August 1983.

5. Brady, Michael, Jean Ponce, Alan Yuille, and Haruo Asada, "Describing surfaces," Joint U.S. France NSF-CNRS Workshop on Robotics, Philadlephia, November 7-9, 1984.

6. Brooks, Rodney, "Symbolic reasoning among 3-D models and 2-D images," *Artificial Intelligence*, vol. 17, pp. 285-349, 1981.

7. Faux, I. D. and M. J. Pratt, *Computational geometry for design and manufacture*, John Wiley, New York, 1979.

8. Fisher, R. B., "Using surfaces and object models to recognize partially obscured objects," *Proc. IJCAI 83*, pp. 989-995, Karlsruhe, August 1983.

9. Grimson, W. E. L., *From images to surfaces: A computational study of the human early visual system*, MIT Press, Cambridge, 1981.

10. Grimson, W. E. L. and Tomas Lozano-Perez, "Model based recognition and localization from sparse three dimensional sensory data," A.I. memo 738, M.I.T. A.I. Laboratory, Cambridge, August 1983.

11. Harmon, Leon, "Automated tactile sensing," *Int. Journal of Robotics Research*, vol. 1, no. 2, pp. 3-32, Summer 1982.

12. Hilbert, D. and S. Cohn-Vossen, *Geometry and the imagination*, Chelsea, New York, 1952.

13. Hillis, W. D., "A high resolution imaging touch sensor," *Int. Journal of Robotics Research*, vol. 1, no. 2, pp. 33-44, Summer 1982.

14. Hu, Ming-Kuei, "Visual pattern recogniton by moment invariants," *IEEE Transactions on Information Theory*, vol. IT-8, pp. 179-187, February 1962.

15. Kinoshita, G., S. Aida, and M. Mori, "A pattern classification by dynamic tactile sense information processing," *Pattern Recognition*, vol. 7, pp. 243-250, 1975.

16. Marr, David and Ellen Hildreth, "Theory of edge detection," *Proc. Royal Society of London Bulletin*, vol. 204, pp. 301-328, 1979.

17. Nevatia, R. and T. Binford, "Description and recognition of curved objects," *Artificial Intelligence*, vol. 8, pp. 77-98, 1977.

18. Nitzan, D., "Assessment of robotic sensors," *Proc. 1st International Conference on Robot Vision and Sensory Controls*, Stratford-upon-Avon, UK, April 1-3, 1981.

19. Oshima, M. and Y. Shirai, "Object recognition using three dimensional information," *IEEE trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, no. 4, pp. 353-361, July 1983.

20. Overton, K. J., "The acquisition, processing and use of tactile sensor data in robot control," Ph.D. Dissertation, University of Massachusetts, Amherst, May 1984.

21. Ozaki, H., S. Waku, A. Mohri, and M. Takata, "Pattern recognition of a grasped object by unit vector distribution," *IEEE trans. n Systems, Man and Cybernetics*, vol. SMC-12, no. 3, pp. 315-324, May/June 1982.

22. Potmesil, Michael, "Generating three dimensional surface models of solid objects from multiple projections," IPL technical report 033, Image Processing Laboratory, RPI, Rensselaer, October 1982.

23. Rock, Irvin, *The logic of perception*, MIT Press, Cambridge, 1983.

24. Rosenfeld, A. and Emily Johnston, "Angle detection on digital curves," *IEEE Transactions on Computers*, vol. C-22, pp. 875-878, 1973.

25. Shapiro, Linda, J. D. Moriarty, R. Haralick, and P. Mulgaonkar, "Matching three dimensional models," *Proc. of IEEE conference on pattern recognition and image processing*, pp. 534-541, Dallas, TX, August 1981.

26. Shneier, M., S. Nagalia, J. Albus, and R. Haar, "Visual feedback for Robot Control," *IEEE Workshop on Industrial Applications of Industrial Vision*, pp. 232-236, May 1982.

27. Tomita, Fumiaki and Takeo Kanade, "A 3D vision system: Generating and matching shape descriptions in range images," *IEEE conference on Artificial Intelligence Applications*, pp. 186-191, Denver, December 5-7, 1984.

28. Unimation,, *User's guide to VAL-II*, Unimation Inc., Danbury, April 1983.

29. York, B., "Shape representation in computer vision," Ph.D. dissertation, University of Massachusetts, Amherst, 1981.