This paper appears in: International Conference in Pattern Recognition, 2002

# Robotic Vision: What Happened to the Visions of Yesterday?

Avinash C. Kak Guilherme N. DeSouza
Robot Vision Laboratory
1285 EE Building, Purdue University
West Lafayette, IN 47907-1285
{kak, gdesouza}@ecn.purdue.edu

## Abstract

We will examine the hopes and dreams of a decade ago when there was a lot of excitement in the community about what could be accomplished with model-based vision in robotics. Many of those dreams have not yet materialized because of a lack of availability of "vision" models (as opposed to CAD models), difficulty of constructing vision models, inadequate robustness of model-based recognition and pose-estimation schemes, excessive sensitivity of recognition and pose-estimation schemes to variations in ambient illumination and object surface condition, etc. Against a background of these difficulties, we will chart out the progress that has been made in the areas of visual servoing, bin-picking, mobile-robot navigation, etc.

## 1 Introduction

Twenty years ago, there was a lot of excitement in the air about what could be accomplished by model-based vision in robotics. Robotics itself was a hot area then, as was the area of artificial intelligence.[1] People at that time generally believed that computer vision, along with the tools of AI, could be used to endow robots with superior intelligence – intelligence at a level that would revolutionize industrial automation. Those of us who were in the vision community believed in particular that model-based vision would indeed solve the object recognition and localization problems in robotics.

That so much was held in store by model-based vision is not too hard to grasp. It seemed so natural to believe that if we had a model of an object, we could form its projections to make a list of visually prominent cues that could then be used to form an initial hypothesis about the presence or the absence of the object in a scene. For a detected cue in an image, the verification of the hypothesis could be carried out by searching for the rest of the structure in the projection.

This approach seemed like such a sure bet that it was difficult to not believe in it. It seemed to be a surer bet than the expert systems for simulating human reasoning and decision making in a computer. All we needed to get this "paradigm" to work was the models of objects that a robot needed to recognize and locate. But since the area of CAD was also in much news during the same period, acquiring models meant simply using the CAD files for the objects.

So why did this vision of the future of robot vision not pan out? Why did model-based vision stumble in the market place of ideas? Why did it not cause a revolution in industrial automation? Why is it that in the kinds of areas we all work in, we have ideas that seem so promising one minute, only to seem so devoid of promise the very next? Robot vision is not the only area that has witnessed this rise and fall of promise. This is also the story of neural networks, AI,

[1] We still recall reading in NYT about a poll of some prominent researchers of the day that indicated their belief that the next big excitement in research lay in solving the mysteries of cognition and perception through AI.

computing with DNA molecules, optical computers, room-temperature superconductivity, quantum computing, etc.

It is important to explore the reasons behind the failures so that we as researchers do not become too cynical about our profession. We need to convince ourselves that while progress in science may indeed be driven by exaggerated expectations, as Lotfi Zadeh once said, the expectations that we create are not so exaggerated so as to be misleading.

Toward that end, in the rest of this paper, we will first highlight some of the more prominent laboratory successes of the past couple of decades that resulted in high excitement about the potential of robotic vision. We will show the kinds of scenes and objects that could be interpreted automatically using model-based vision. Since our own data is readily accessible to us, we will make our points using the images that came out of our own laboratory, but our statements apply to the research community at large. Next we will discuss why these initial experiments failed to scale up into real-world applications. This we will do by presenting the failure modes of the earlier research. Finally, we will briefly survey some of the current research in robotic vision that is more likely to have immediate societal payoffs than the earlier laboratory advances. We must hasten to add that many of the lessons learned from the earlier laboratory successes are incorporated in the current research pursuits of the community.

# 2 The Initial Promise

The fact that model-based vision has not lived up to its initial promise should not be construed to imply that the initial promise was based entirely on hype. On the other hand, the initial promise was based on some extremely impressive results – or, at least, results that appeared to be extremely impressive at that time. In this section, we will briefly go over the kinds of objects and scenes that the computer vision systems of the day could understand without too much difficulty. The objects and scenes that we will show could not have been handled by the vision systems of the sixties and the seventies. We will show

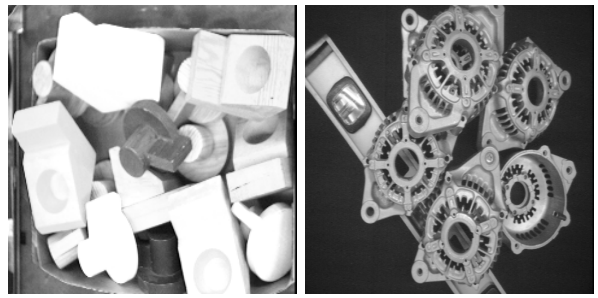


Figure 1: *Successful laboratory computer vision systems that worked well for the types of scenes depicted here led initially to considerable excitement about the potential of robot vision in advanced automation. Scenes like the one shown on the left could be handled with range mapping, and the one on the right with binocular stereo.*

these objects and scenes in two different contexts: bin-picking of some fairly complex parts in complicated jumbles, and indoor mobile robot navigation.

## 2.1 3D Object Recognition and Localization

The initial promise in bin-pickings kinds of applications was based on the design of computer vision systems that could successfully interpret scenes like those shown in Figs. 1(a) and 1(b).

Successful computer vision systems for scenes such as the one shown in Fig. 1(a) used range imaging for sensing and object models that fully described the geometry of the exterior surfaces. Range mapping of such scenes with structured-light sensors resulted light stripe images like the one shown in Fig. 2(a). When this data was subject to low-level processing and region based segmentation, one obtained segmentations like the one shown in Fig. 2(b). As mentioned already, interpretation of this data was carried out with the help of models that fully described the exterior surfaces of the objects.

Equally exciting results in object localization were obtained with binocular stereo – for the kinds of objects and scenes shown in the Fig. 1(b) above. We show in Fig. 3 a pair of stereo images of such a scene

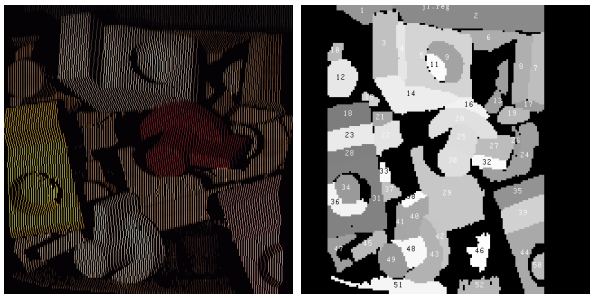


Figure 2: *(a) Data from Structured-light Sensor and (b) Segmentation output for MULTI-HASH.*

at the top and some results of intermediate processing in the rows below. Shown in the third row are the primitives that correspond roughly to the elements of the vision model for the objects in the scene. Objects were recognized by subjecting these primitives to verification on the basis of the presence of other primitive at specific distances and with specific angular intervals between them. Given the stereo nature of the data, the relational constraints between the primitives were enforced in 3D and the objects localized in 3D.

We believe that the stereo based approach described above succeeded primarily because of the simplicity of the vision model for the objects. The model for each object consisted of a large circle surrounded by four small holes. For model based matching, all we had to do was to extract those regions from the image that satisfied roughly the area and circumference properties for the large circle in the center of the object. The small surrounding holes then served as a means for verification. The simple model used is depicted in Fig. 4.

## 2.2 Vision-Based Mobile Robot Navigation

The early promise in vision for indoor mobile robots was based on the success of navigation in the kinds of cluttered up hallways depicted in the Fig. 5. Obviously, if a robot could figure out where exactly it was in a hallway despite the obscurations shown in the figure, was there anything the robot couldn't do?

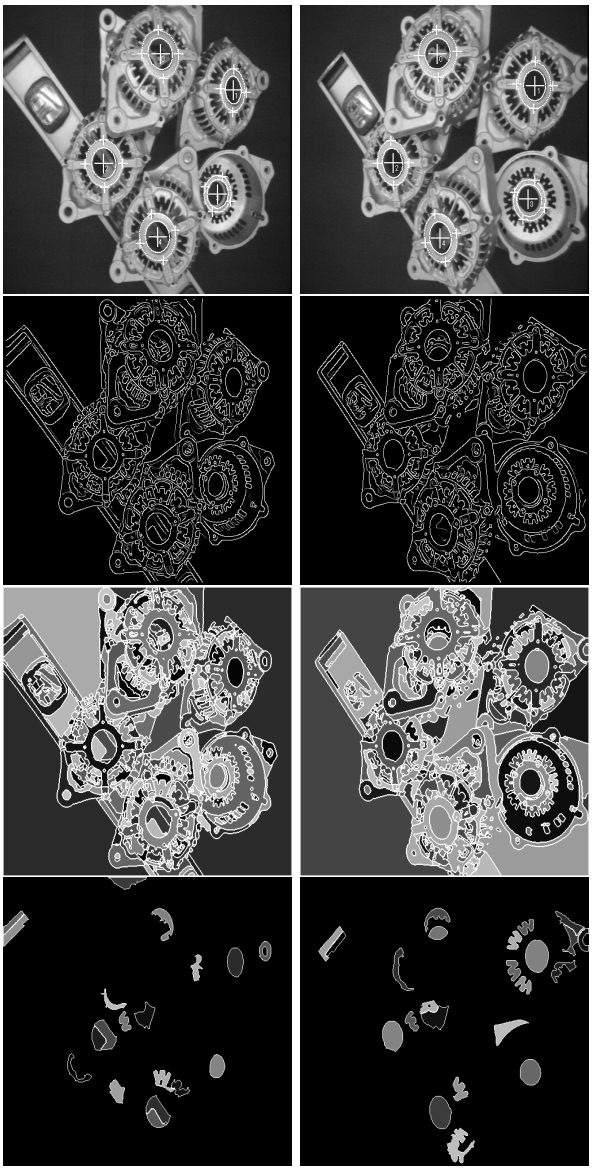


Figure 3: *The top row shows a stereo pair of images.; the row below the edge maps for the stereo images; the third row region-based segmentations that are driven by the edges shown in the second row. From this segmentation, only those primitives are retained that correspond approximately to the components in the vision model for the objects. The retained primitives are shown in the last row. A white cross is drawn in the middle of an object in the first row if the object was localized by the computer with sufficient accuracy.*

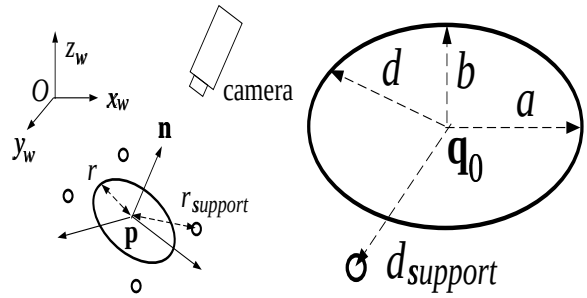



Figure 4: *Model used for the localization of the objects shown in Fig. 3.*

Figure 5: *Some of the intermediate results in FINALE's localization algorithm. (a) the camera image and the superimposed expectation map; (b) output of the model-guided edge detector; (c) uncertainty regions associated with the ends of the model line features; and (d) matching after Kalman filtering.*

But, alas, as it turned out, there indeed was (but we will get to that in the next section). Vision based hallway navigation experiments involving the kinds of scenes shown in Fig. 5 were truly thrilling because they showed that, with proper top-down control, a robot could be made to be blind to the huge number of low-level features – most irrelevant to the task of navigation at hand – that would be produced by a low-level feature detector. In the past, complex camera images such as the one shown in the figure used to be formidable to deal with because of an inordinate number of features that would be output by any feature detector. This difficulty was compounded by the subsequent need to determine whether or not any subset of these features matched robot expectations. But now, using a model-based framework such as that of FINALE, a robot only needed to examine those portions of the camera image that contained low-level features in the "vicinity" of model features, the extent of the "vicinity" being determined by the uncertainty in the position of the robot.

Systems that could perform reliable vision-based navigation in the kinds of hallway environments shown in the figure used geometrical models of space and probabilistic models of the uncertainty in the position of the robot. The models of uncertainty, instantiated through extensive experimentation, were used to drive a model based Kalman filter for establishing correspondences between the features in an expectation map and a camera image. A robot would construct an expectation map from its geometrical representation of the interior space and the Kalman filter would try to reconcile the features in the expectation map with the features in the image. Shown at top left in Fig. 5 is a camera image on which is superimposed an expectation map that the robot has rendered from the geometrical model of the space assuming that the robot is at the center of the uncertainty region. The image at the top right shows the low-level features extracted from the camera image. Note that these features are extracted only in the "vicinity" of the features in the expectation map – that's how the robot keeps itself from getting overwhelmed by what would otherwise be an ocean of edges extracted from the image. The ellipses in the image at the bottom left show the uncertainty regions

associated with the end points of the vertical model edges. Convex hull of these ellipses for the two ends of a model edge is the part of the image that is analyzed for that model edge. The image in the bottom right displays the correspondences achieved by the model based Kalman filter. In its latest incarnation, FINALE can navigate at an average speed of 17 m/min using an ordinary PC-based architecture (Pentium II 450Mhz, with no special signal processing hardware), and its self-localization routine, which helps the robot figure out where it is using vision, runs in less than 400ms per image.

Early promise accomplishments in vision for indoor mobile robotics also included systems that used topological representations of space and fuzzy-logic based representations for robot position uncertainty.

### 2.3 Appearance Based Approaches

During the nineties, as unease began to develop that laboratory successes of the mix-and-match variety of feature-based approaches may not scale up to the real world, there came along the appearance based "paradigm". The basic idea here was simply to let the set of all the appearances of an object serve as its model. While, as with the traditional-style mix-and-match approaches, the initial laboratory successes were stunning, it didn't long to see that, at least from the standpoint of general purpose computer vision, the appearance based approach was more brittle than the feature based approaches. For an appearance based approach to work, greater care was needed for conditioning the data before matching it with the "model". By conditioning we mean normalization, centering, etc. Nonetheless, this remains a favorite approach today for important applications such as face recognition.

## 3 Failure Modes and Limitations

We believe it is useful to dwell on the failure modes of the experiments we described in the previous section to get a sense of why such experiments failed to cause a revolution in the world at large.

Focusing first on the experiments we showed for bin-picking, the problems that affect the robustness of systems that use range data for object recognition and localization include: Sensitivity of range data segmentation to surface orientation, surface reflectance, depth ambiguity in the vicinity of jump edges, etc. And the problems that beset systems that are based on 2D imaging, stereoptic or otherwise, are all of the usual ones – problems with low-level feature extraction, excessive sensitivity of low-level feature extractors to illumination and surface reflectance, specularity effects, etc.

We will now give the reader a quantitative sense of the problems encountered when range data is used for object recognition and localization. In one experiment we used our MULTI-HASH system on ten random scenes similar to that in Fig. 1(a). In seven of those scenes, MULTI-HASH successfully recognized at least one object. Experiments showed that there were two root causes for this 30% failure rate: improper surface segmentations, and the occasional inability to extract a vertex-centered local feature set (LFS) needed for the formation of object and pose hypothesis. In a random bin, it was not uncommon for the objects to present themselves in such a way that the vision system would detect no local feature sets at all.

While the first type of errors could easily be blamed on the sensors used (and could therefore be alleviated by better, meaning commercial-grade, sensors), the second type of error was more fundamental to the design of the vision systems. Basically, this error resulted from the choice of features for the formation of object identity and location hypotheses. Most researchers used convex vertices along with their surrounding surfaces as local feature sets (LFS) for the formation of hypotheses. But that creates a problem for objects that do not present to the camera a sufficiently large collection of such feature sets for hypothesis formation. Obviously, if an object is completely curved, it will have no such local feature sets. So the success of such experiments was based on an assumption that could not be easily extended to more general objects.

Therefore, it would be fair to say that the primary failure modes – meaning failure modes that cannot be

easily rectified by just improving a sensor or even by improving low-level feature extraction – were consequences of what seems in hindsight as experimental expediency or an implementation shortcut. When that research was first carried out, there did not seem to be anything disrespectable in the assumptions incorporated in those implementation details. But, with the passage of time, it became clear that the problem of hypothesis formation in scenes was probably just as difficult as the problem of object recognition.

Moving on to stereo based systems for bin picking, of the kind that have worked successfully on the types of images shown in Fig. 3, how come these laboratory demonstrations did not cause a sensation in the industry? In hindsight it seems that the primary reason for the success of these experiments lay in the simplicity of the vision models for the objects. The simpler a vision model, the more efficient the search for the features that help the vision system recognize and locate objects. What also helped in these experiments was that the scenes did not contain a variety of objects – only one type was involved. So the vision model used did not have to possess attributes for discrimination against other possible objects.

We must say that the assumption that the scene has objects of only one kind is not too limiting from a practical standpoint, since such is often the case in industrial situations where the goal is more to localize a known object than to identify an object. But whether or not an object can be described by a simple vision model is a major issue. We suspect that a large class of industrial objects could be described by simple vision models, but we still do not have a reliable means for the construction of such models.

For model-based vision for indoor mobile robot navigation, the failure modes very much parallel those for the bin-picking case. While many failures are indeed caused by the peculiarities of the sensors, those are not the fundamental failures in our opinion. As was the case with the limitations created by the excessive simplicity of the local feature sets for bin picking, the fundamental limitations of the work done so far in mobile robotics vision are caused by our inability to specify simple vision models for interior space. Models that are excessively rich in geometrical details simply do not scale up and can at best be used for incremental localization, as opposed to global localization.

# 4 The Current Excitement

In this section, we will list some of the topics of robot vision that the authors (and we believe the research community at large, too) are excited about. Most of these, especially those that have a human in the loop, are borne out of the realization that fully automated approaches to model-based vision will not be feasible for a long time to come.

## 4.1 First a Caveat

This section only includes a subset of those aspects of computer vision that we believe are relevant to robotics. The specific topics chosen reflect the excitement of the authors. Although we believe that a significant number of researchers in robotics would agree with what we have said in this section, we are open to the possibility that many other respectable researchers might construct a list of topics different from ours.

## 4.2 Visual Servoing

While it has not yet been possible to show that the impressive laboratory demonstrations of model-based systems for indoor mobile-robot navigation can scale up to the real-world, we believe that the same ideas of model-based incremental localization can be used with greater likelihood of success in the real-world applications of visual servoing. A real-world application of mobile robot navigation can call for a vision system to operate under widely (and sometimes wildly) varying conditions. On the other hand, in visual servoing for applications such as assembly-line tracking in a factory, the scene to be servoed on stays more or less the same during the entire tracking process.

To illustrate more pictorially what we mean by visual servoing for line tracking, shown in Fig. 6 is a wheel hub suspended from a conveyer in an automobile assembly line. The goal here is for an arm robot

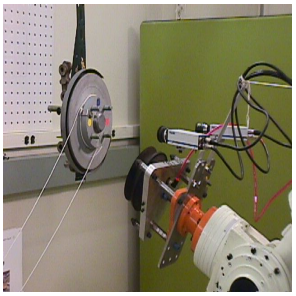


Figure 6: *Robot arm tracking a wheel hub. The two stereo cameras mounted on the end-effector are for fine motion control. A separate camera mounted on the ceiling (not shown) is for coarse control.*

to track the hub visually with the help of two stereo cameras mounted on its end-effector while carrying out assembly operations on the hub. A typical assembly in this situation would consist of installing a wheel on the hub, threading nuts onto the bolts, and tightening the nuts.

The main challenge in this real-world application of visual servoing is posed by the fact that the wheel hub may occasionally go out of the tracking range of the two cameras on the end-effector. So an important challenge becomes the recapture of the object to be tracked.

We have developed a two-loop approach to solve this problem. A ceiling mounted camera is used to keep track of the hub on a global basis. As far as this camera is concerned, a hub is a large blob that needs to be segmented out from the background using color, texture, and motion cues. As the hub (or an entire vehicle if the hub is already assembled with the rest of the vehicle) gets sufficiently close to an assembly station, the control function is taken over by stereo cameras mounted on the robot end-effector. This control is in 3D and in real-time, meaning that all the three-dimensional pose parameters of the hub are calculated at frame rate. We refer to the control loop accomplished with the ceiling-mounted camera as *Coarse Control* and to the control loop achieved by the robot-mounted cameras as *Fine Control* (Fig. 7).

The variables that are relevant to the Fine Control

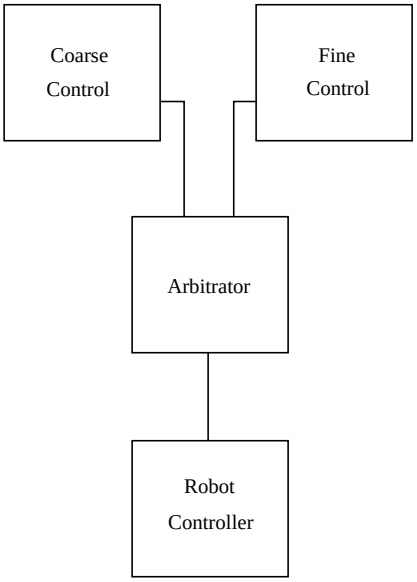



Figure 7: *A control architecture for visual servoing in the real world. When the object to be tracked goes out of the visual range of Fine Control, the Arbitrator hands the control over to Coarse Control. And when the object is again within the capture range of the stereo cameras, Arbitrator hands the control over to Fine Control.*

algorithm are : 1) the time to process visual cues in order to obtain a sufficiently accurate 3D-pose estimation of the target; 2) the time it takes to compute the new coordinates of the object; 3) the time it takes to predict the future motion of the object based on its current coordinates, its previous coordinates, the noise in the system, the dynamics of the object, and the dynamics of the line; 4) the control law used for visual servoing; etc. The visual servoing performed by the Fine Control runs at close to 50 frames per second.

The job of Coarse Control is to detect the object and initiate the motion of the robot towards the object. This motion does not need to be fast or accurate. All that is required from Coarse Control is to position the robot so that the Fine Control cameras attached to the robot end-effector can effectively see the target object. Once the target is in the field of view of the stereo cameras, the Control Arbitrator can switch mode, assigning the control over the robot to the Fine Control.

## 4.3 Human Assisted Feature Delineation

Robot vision could be applied in many more applications if somehow the segmentation problem could be solved. Obviously, robot vision is much more than just segmentation, but reliable segmentation is the first key step in many applications. Fortunately, there exist important applications in which it is possible for a human to intervene and help out with the segmentation step. A hallmark of these applications is that it's alright for a few seconds or even a few minutes to elapse between the time an image is pulled up on a screen and the time when processing results on the image should become available. To cite a non-robotic example, when a physician examines a single radiological image (or a small set thereof) on his/her workstation, the physician does not require that the image(s) be processed at video rate. It's alright if the computer took a few seconds to come back with its results. In such applications, it makes sense to ask the human in the loop to help out with those aspects of image processing that are still beyond the capabilities of computers.

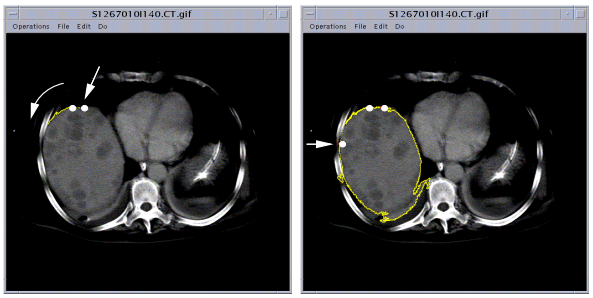



Figure 8: *(a) The two points on which a human clicked initially are shown by two white dots; and (b) the entire boundary of the liver was extracted by clicking at just one more point, as shown here.*

Exactly such a system is being developed in our laboratory by Christina Pavlopoulou for human assisted segmentation of images. In the system being developed by Pavlopoulou, the human clicks on just a couple of images on the boundary of the liver. The computer develops a Bayesian model for the pixels on the two sides of the perceived object boundary. The computer uses this model to extend the object boundary to the maximum extent possible. The chances are, of course, high that such a boundary started with just a couple of human-entered points would go awry somewhere. When that happens, all that the human is called upon to do is to click on where a correction to the boundary is to be made. In this manner, the entire boundary of a fairly complicated looking object can be extracted with just a few human clicks, usually fewer than six.

Fig. 8 shows such a boundary obtained with the help of the clicks entered at the locations marked by white dots. Even though the image shown is from the medical domain, the same ideas would extend to a more industrial application.

## 4.4 Human Assisted 3D Model Acquisition

Model based robotic vision obviously needs vision models for objects. It is now widely believed that it is best to make vision models using the same sensors that would later be used for object recognition

and localization.

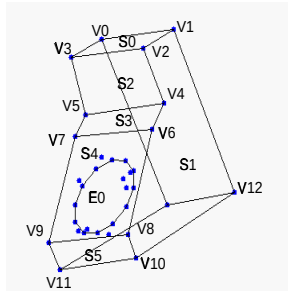
Our laboratory is developing two different types of vision models: those that are made with 2D vision using multi-viewpoint stereo for the gathering of the 3D coordinate information; and those that are made directly from range maps. The human plays an indispensable role in the construction of both these types of models.

Human-assisted model construction using 2D vision is being developed in the lab by Yuichi Motai. The human places an object under a robot-held camera in its various stable poses. The robot takes five images of the object, one from directly overhead and others from four other viewpoints. The human interaction here consists of

- Helping the computer with the mid-level grouping of low-level features. For example, for polyhedral features, the human clicks on the vertices and helps the computer put together polygonal shapes. For curved features, such as elliptical features, the human clicks on low-level processed images to supply the eight points needed for ellipse reconstruction.
- Helping the computer establish stereo correspondences between the five images in each pose of the object. For each candidate point in the central view, the computer tries to help out the human by drawing an epipolar line in the image in which the candidate is sought.
- Helping the computer establish pose-to-pose correspondences between the 3D features extracted in each pose.

An important lesson we learned in this research was that the humans can commit all sorts of errors when helping a computer with the above three tasks. So the formulation of human-friendly interaction protocols and the detection of the errors committed by the humans are central to this kind of research.

Shown in Fig. 9 is the wireframe representation of a vision model that was constructed in the manner described here. Associated with each surface and each edge in the model is a data structure that contains numerical values for the various attributes relevant to that feature.




| Feature Surfaces | Vertex Entities | Area [$mm^2$] | Perimeter [mm] | Shape Complexity |
|---|---|---|---|---|
| Polygon S0 | V0 V1 V2 V3 | 3045 | 244 | 4.42 |
| Polygon S1 | V1 V2 V4 V6 V8 V10 V12 | 8769 | 441 | 4.71 |
| Polygon S2 | V3 V2 V4 V5 | 3828 | 262 | 4.23 |
| Polygon S3 | V4 V6 V7 V5 | 1596 | 212 | 5.32 |
| Polygon S4 | V6 V8 V9 V7 | 5005 | 542 | 7.66 |
| Polygon S5 | V8 V10 V11 V9 | 1144 | 202 | 5.97 |
| Ellipse E0 | 8 cardinal points | 2827 | 188 | 3.56 |
| Vertices | Adjacent Vertices | X Coordinates | Y Coord. | Z Coord. |
| V0 | V1 V3 V12 | 36.9 | 66.6 | 124.9 |
| V1 | V0 V2 V12 | -46.4 | 35.3 | 125.1 |
| V2 | V1 V3 V4 | -44.8 | 1.7 | 125.5 |
| V3 | V0 V2 V5 | 39.3 | 10.9 | 125.1 |
| V4 | V2 V5 V6 | -42.7 | -3.4 | 83.5 |
| V5 | V3 V4 V7 | 47.7 | 8.4 | 84.2 |
| V6 | V4 V7 V8 | -42.1 | -19.5 | 74.0 |
| V7 | V5 V6 V9 | 45.8 | -8.8 | 71.6 |
| V8 | V6 V9 V10 | -31.6 | -81.0 | 18.2 |
| V9 | V7 V8 V11 | 56.0 | -67.5 | 15.4 |
| V10 | V8 V11 V12 | -24.9 | -83.1 | -1.6 |
| V11 | V9 V10 V13 | 52.6 | -66.6 | -3.6 |
| V12 | V1 V10 V13 | -30.9 | 37.0 | -1.9 |

Figure 9: *This vision model was constructed with a human helping the computer: (a) wireframe representation of the model; and (b) the numerical attributes associated with the various features of the model.*

# 5 Conclusions

It appears that high-technology areas, robotic vision being one of them, are prone to excessively heightened expectations created by laboratory demonstrations that appear stunning at the beginning, but not so in hindsight. The assumptions embedded in the original laboratory work, often a consequence of the imperatives of giving a sound mathematical footing to a piece of work, appear much too simplistic when the same experiments are tried in real-world scenarios.

This paper narrates this *initial-high-excitement-followed-by-disappointment* story of robotic vision. We start out by listing the laboratory successes of the last two decades that were truly stunning and represented major advances in the state-of-the-art at that time. We then go into why it has not so far been possible to translate those laboratory advances into more efficient systems and processes for industry.

We end our paper by listing a small subset of robotic vision topics that are more likely to be successful. Many of these, offshoots of the early successful laboratory demonstrations of robotic vision, are more likely to scale up to the real world because they are founded on more realistic assumptions. Some of them bring a human into the processing loop to help out with tasks that the computer are ill-equipped for.