COMPUTATIONAL VISION

H. Barrow

Rapporteurs: Mr. K. Heron
Dr. F. Panzieri

Abstract

Vision is undoubtedly man's most important sense. If a high-performance, general-purpose machine vision system could be developed, the range of applications would be immense. Considerable effort has been directed towards automating image analysis, and some useful techniques and systems have been developed for the more constrained and well-defined tasks. Developing general-purpose artificial vision systems to deal with less predictable and less structured scenes, however, has proved surprisingly difficult and complex.

Research in the last few years has begun to uncover some fundamental computational principles underlying vision that apply equally to artificial and natural systems. By designing systems in accordance with these principles, it appears possible to alleviate many of the difficulties that plague current computer vision systems, and to which the human visual system is largely immune.

In these lectures a computational view of visual perception is presented. Some of the early work in the field will be discussed in this context and current thoughts on the overall organization and operation of a general-purpose visual system put forward. Some contemporary computer vision systems will be discussed in the light of this computational model of vision.

Introduction

Vision is undoubtedly man's most important sense. It enables him to find his way about the world, to recognize and manipulate objects, to understand the actions of others, and generally to gather the information needed to attain his goals. All this can be accomplished at a safe distance, without direct physical involvement, or even giving away his presence.

Needless to say, if a machine vision system with near-human performance could be developed, the range of applications would be immense: making maps, inspecting industrial parts, interpreting medical x-rays, screening tissue cultures, and analyzing weather satellite imagery are but a few examples. Considerable effort has been directed towards automating image analysis, and some useful techniques and systems have been developed for the more constrained

and well-defined tasks. Developing general-purpose artificial vision systems to deal with less predictable and less structured scenes, however, has proved surprisingly difficult and complex. This has been particularly frustrating for vision researchers, who daily experience the apparent ease and spontaneity of human perception.

Research in the last few years, however, has begun to uncover some fundamental computational principles underlying vision, that apply equally to artificial and natural systems. These principles help us to understand the limitations of early machine vision systems and lay a foundation for building future systems capable of high performance in a broad range of visual domains.

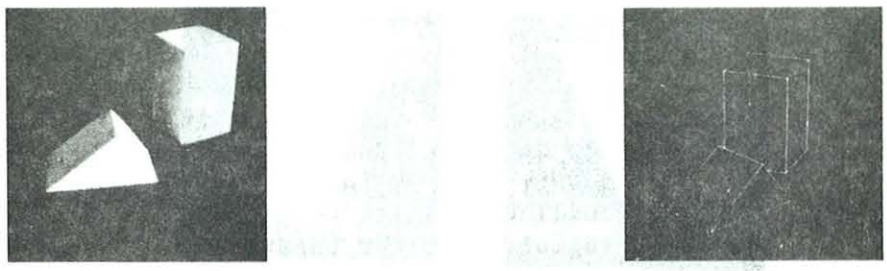
Historical Development

A major computational principle of vision is that competence depends on the models available. The earliest computer image processing systems attempted to perform a specific task, using a minimum of image description and modelling. Extreme examples are the many correlation matching systems that have been developed for tasks such as target detection, change detection and stereo mapping. The input consisted of two image arrays: a sensed image and a reference image. The output was another array representing degree of match, degree of change or terrain height. Since correlation matching makes no use of imaging models, it has no basis for comparing images obtained under different viewing conditions. Consequently, target detection was limited to a rigidly constrained class of object, representable by a pictorial template, observed from a specific viewpoint. Similarly, change detection required reconnaissance images obtained from the same viewing angle, at the same time of day and the same season. Stereo mapping worked only for smooth terrain, with viewpoints sufficiently close together that the images were locally almost identical.

To overcome these severe limitations, additional levels of description and modelling were clearly needed. At the very least, iconic models must be discarded and replaced by symbolic abstractions that capture a broader class of object or viewing conditions. A beginning was the attempt to match pictorial features (such as edges and regions) extracted from images with corresponding features predicted from symbolic object models. This approach was taken at several Artificial Intelligence laboratories and was termed "Scene Analysis" or "Computer Vision" to distinguish it from earlier work on "Image Processing".

Early Computer Vision Systems

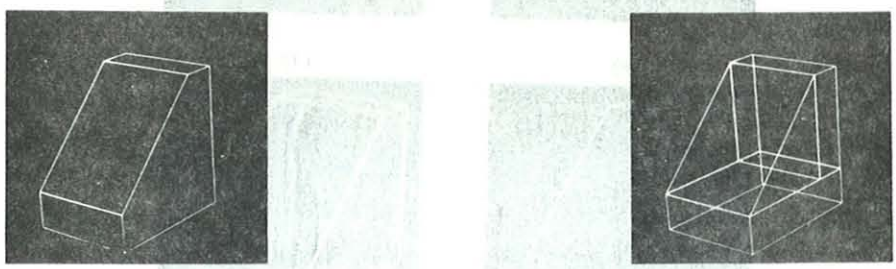
Early Computer Vision research was pursued in simplified scene domains, in which objects and their appearances were easy to model. Perhaps the most popular domain was the "Blocks world", comprised of polyhedral objects with uniformly-colored matte surfaces on a



a) b)

Figure 1 Roberts' program for perceiving Blocks World scenes.

a) Computer display of original picture.
b) Synthetic view from another location.



a) b)

Figure 3 Decomposition into primitive prototypes.

a) Line drawing of object.
b) Decomposition into two bricks and a wedge.

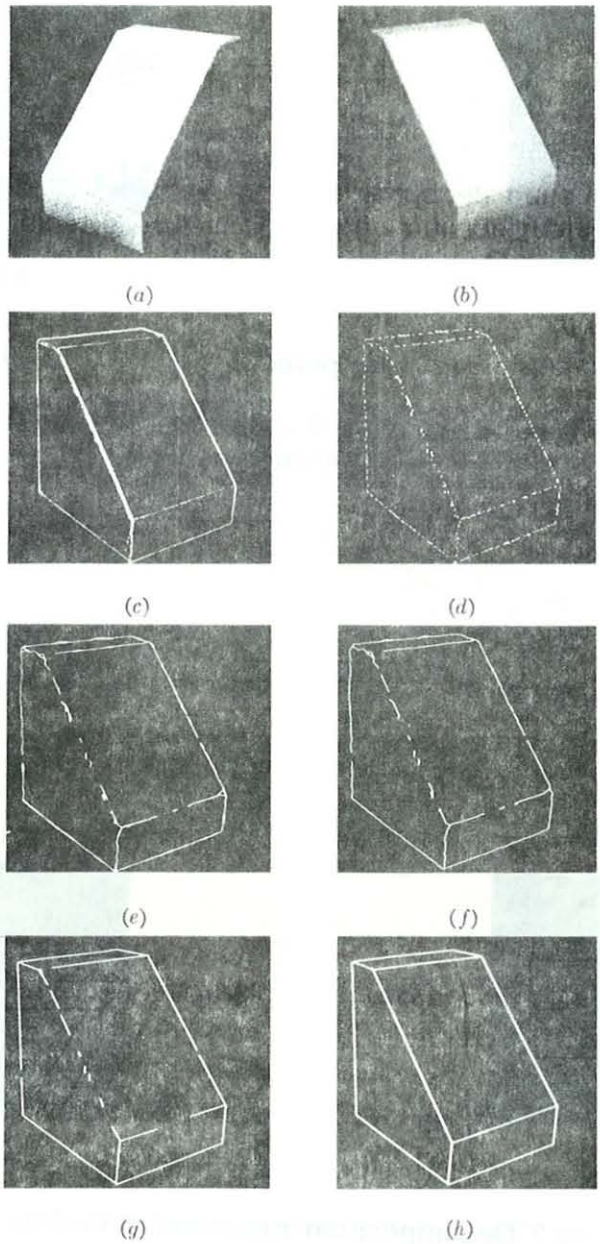


Figure 2 Picture to line drawing.
a) Original picture
b) Computer display of picture (reflected)
c) Differentiated picture
d) Feature points selected
e) Connected feature points
f) After complexity reduction
g) After initial line fitting
h) Final line drawing

table-top. Such objects are easily modelled in three-dimensions by the coordinates of their vertices and a specification of the edges that link them. Ideally, their images are characterized by polygonal regions of approximately uniform brightness, with discontinuities of brightness at their boundaries, corresponding to surfaces and edges respectively. In practice, however, shadows, texture and scattered light cause complications.

Pioneering work in perception of blocks-world scenes was performed by Roberts (1965). His program used an image dissector camera to look at scenes, such as Figure 1a, containing bricks, wedges, hexagonal prisms and objects composed of these primitives. It could determine the dimensions, location and orientation of visible objects and, as a demonstration of its "understanding", it could generate a line drawing of the scene from any other viewpoint (e.g. Figure 1b).

Roberts' program operated by first finding places in the image where brightness, $B_{i,j}$, changed abruptly. Such discontinuities were found using the local operator:

$$G = \sqrt{((B_{i,j} - B_{i+1,j+1})^2 + (B_{i+1,j} - B_{i,j+1})^2)}$$

which yields the magnitude of the brightness gradient, G, at each point. The output of the operator is shown in Figure 2c for the image of 2a. A set of masks was then used to detect short alignments of points of high gradient, corresponding to primitive line elements in one of four orientations (Figure 2d). Isolated elements were discarded and the remainder grouped into line fragments (Figure 2e). Collinear fragments were then merged into lines, and these were segmented at corners (Figure 2g). The final result, after some tidying-up, was a "drawing" of the scene with lines representing surface boundaries and closed regions representing surfaces (Figure 2h).

To interpret the resulting line drawing, regions were classified on the basis of shape, as triangles, quadrilaterals and hexagons, which suggested possible object prototypes (e.g. triangles suggest wedges, etc.). A selected region was then topologically matched to part of the prototype. From the projective geometry of the camera (and the assumption that the object was supported, either by the table top or by a previously recognized object) the position, orientation, and scale of the object could be precisely determined. The camera model was then used to project the complete transformed prototype onto the two dimensional image, to obtain a predicted line drawing (with hidden lines removed). A comparison between the predicted and extracted line drawings provided verification of the hypothesized prototype, accounting for additional lines and regions. The recognized piece of the extracted line drawing was then cut away and the remainder considered, repeating this process until all the detected lines and vertices were explained. Figure 3b shows the

decomposition, into two bricks and a wedge, selected by the program to explain the line drawing of Figure 3a.

Roberts' work was founded upon several major concepts. The program was partitioned into two strictly sequential phases: segmentation, in which the original intensity image was reduced to a line drawing, and interpretation, in which the line drawing was explained using the three-dimensional prototypes and a geometric camera model. Segmentation was viewed as a grouping process based on implicit knowledge of the importance of straight edges. Interpretation used topological features to hypothesize possible matches, and projection to verify a correct match.

Region Analysis

Dependence upon well-defined straight edges and precise geometric models of objects makes it difficult to generalize blocks-world techniques to real scenes. An alternative approach is to partition an image into regions of approximately uniform brightness, corresponding to surfaces. This can be accomplished by initially partitioning the image into elementary regions of constant brightness, and then successively merging adjacent regions for which the contrast across their common boundary is sufficiently low, until only boundaries with strong contrast remain. (See Figure 4 (Brice and Fennema 1965)). Unlike edges linking, "region growing" does not require the assumption that boundaries are straight, and generalizes more readily to characteristics other than brightness, such as texture and color, which are important in natural scenes.

A segmentation of an image into regions can be described in terms of a relational graph whose nodes represent regions and whose arcs represent properties (e.g. size, shape) of and relations (e.g. adjacent to, larger than) between regions, as in Figure 5. Particular views of known objects can be similarly represented and recognition can be accomplished by matching the graphs. A program by Barrow and Popplestone (1971) used this approach to recognize a small set of simple curved objects, such as pencils, eyeglasses and teacups. Descriptions of new object views were acquired by forming relational descriptions from several training images, and computing means and standard deviations for the values of properties and relations.

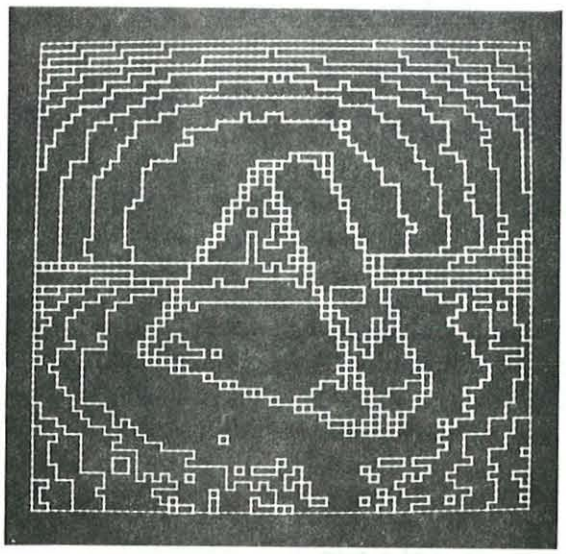
Relational descriptions can capture more than geometrical structure and hence are appropriate for classes of objects whose form may not be precisely specified in advance. However, when based upon features of two-dimensional pictorial regions, as in the early work above, their power is limited by their sensitivity to viewpoint.

Interpretation-Guided Segmentation

A major problem with the sequential program organization (segmentation followed by interpretation) used by both Roberts and

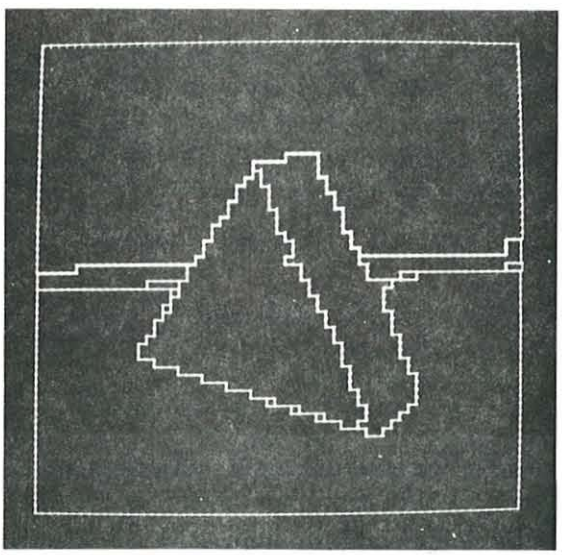
(a)

a) Digitized grey-scale image, 120x120 picture elements.



(b)

b) Initial partition into elementary regions.



(c)

c) Surviving regions after merging.

Figure 4 Region growing

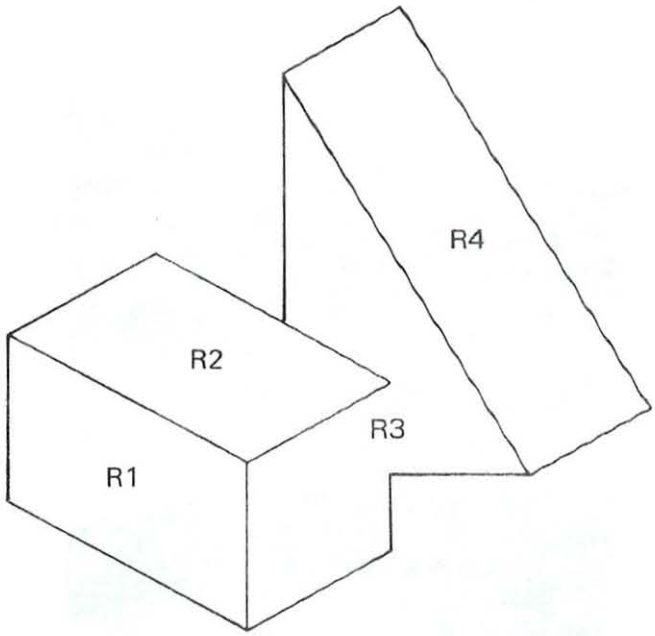


Figure 7 Blocks scene with a missing line.

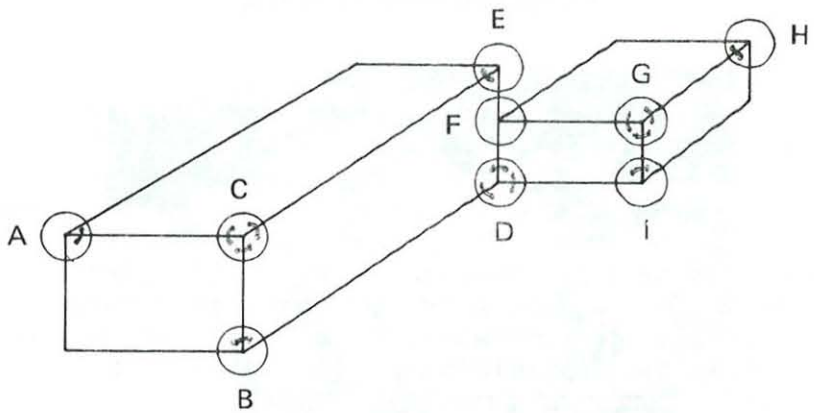


Figure 8 Grouping regions into bodies using local vertex information.

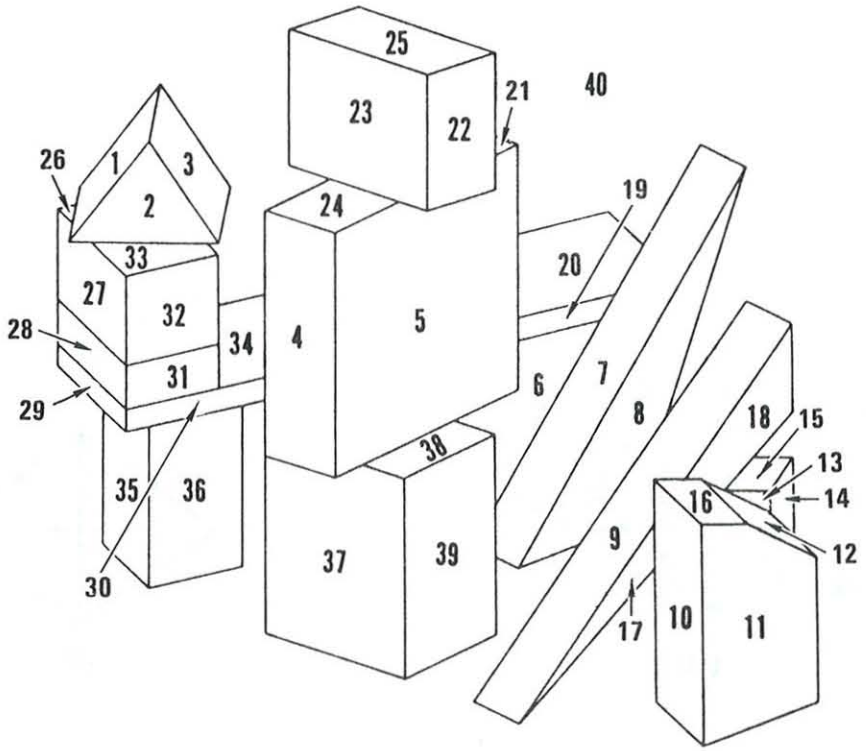


Figure 9 A complex scene successfully processed by Guzman's program.

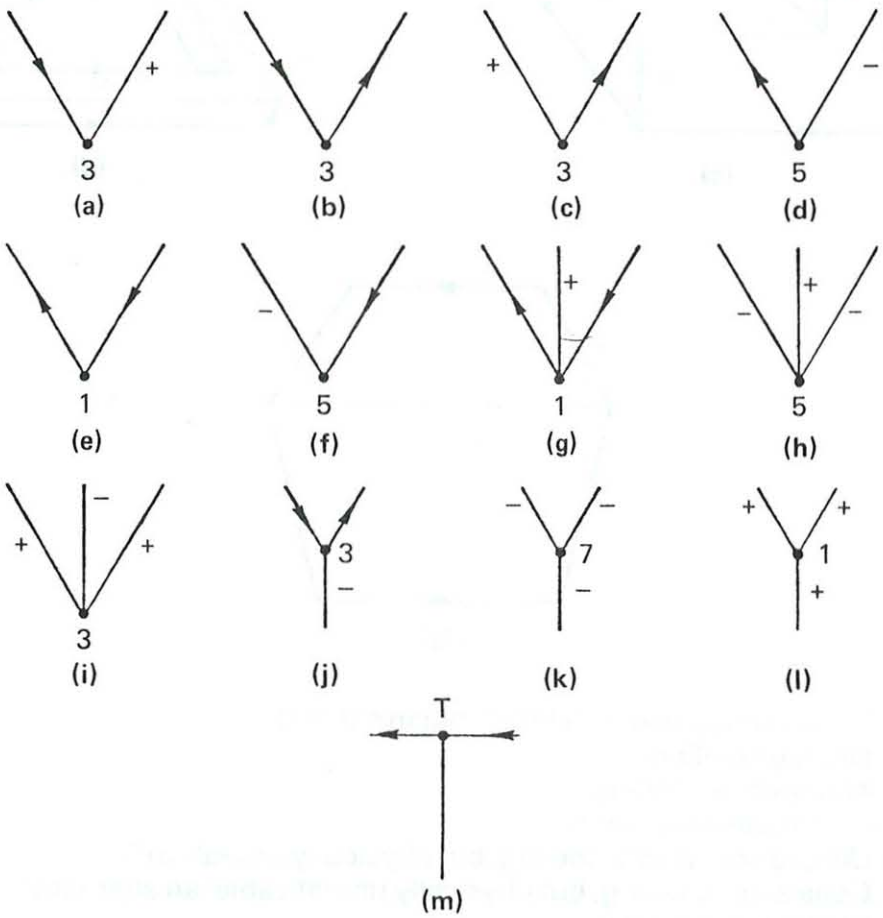


Figure 10 Huffman's catalog of all possible interpretations of the vertices found in images of trihedral solids.

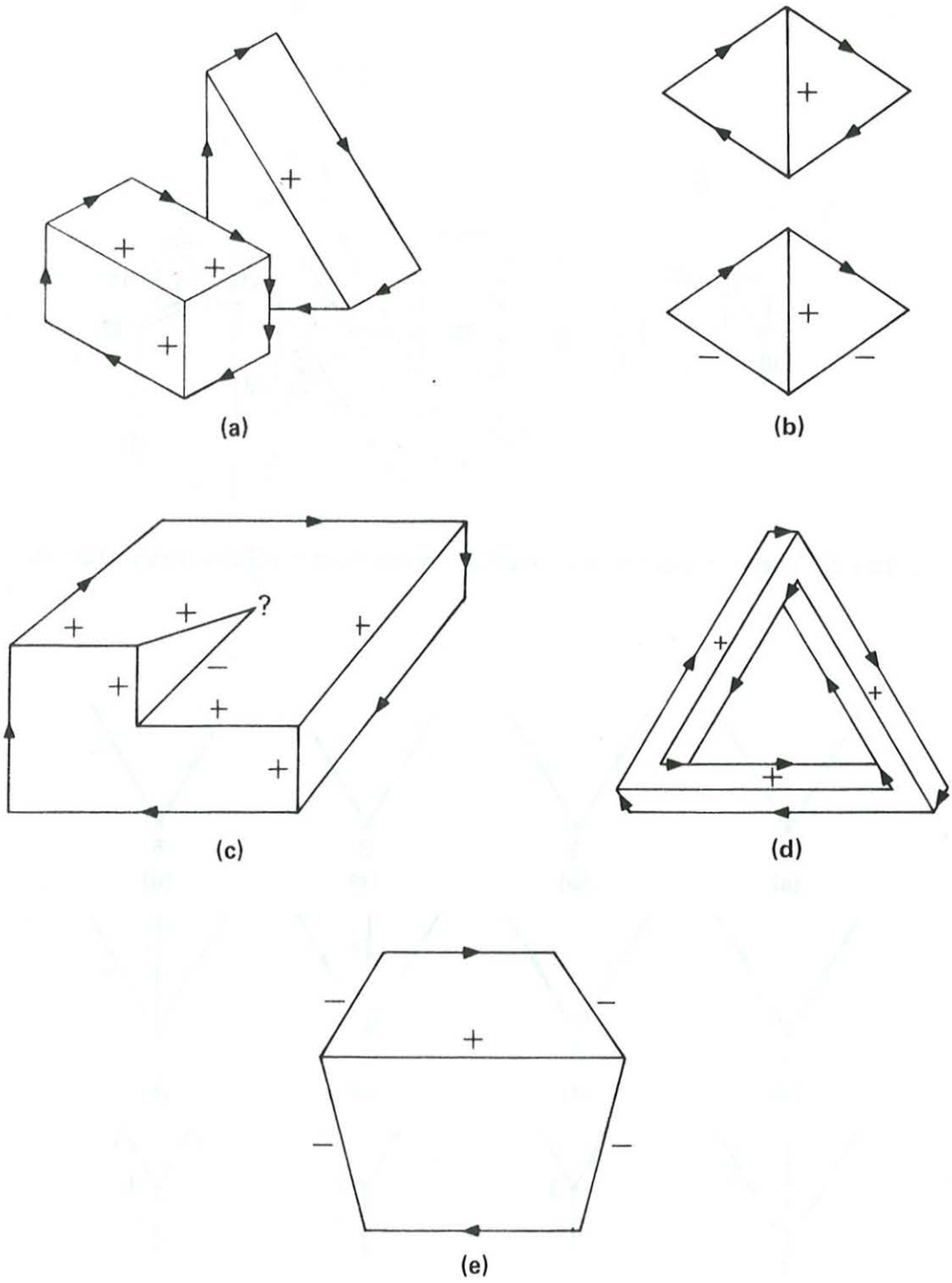


Figure 11 Drawings labeled with the Huffman catalog.
a) Unique labeling
b) Ambiguous labeling
c) Inconsistent labeling
d) Unique consistent labeling, but physically unrealizable
e) Consistent labeling, but physically unrealizable: an alternative, realizable labeling is possible.

features on which Guzman relied : by linking line segments at a vertex, instead of regions, such line drawings as Figure 7 could be handled.

Clowes (1971) and Huffman (1971) independently provided a firm theoretical foundation to account for the remarkable performance of Guzman's simple heuristics. They exhaustively cataloged vertices that could arise in line drawings of trihedral solids (solids whose corners are formed by exactly three meeting edges), and then used the catalog to interpret lines as corresponding to convex or concave solid edges. The crucial finding was that for a given type of vertex, only certain combinations of line interpretations were physically possible (Figure 10). Thus interpretation could be performed by searching for a set of line interpretations consistent with this catalog. The search is restricted by the constraint that every line must be assigned the same label by the junction interpretations at its two ends.

Figure 11a shows interpretation of a simple line drawing according to Huffman's catalog. Note that, as in 11b, drawings often admit more than one consistent interpretation. On the other hand, it is now often possible to detect when a line drawing cannot correspond to a physical object, because no consistent interpretation can be found (Figure 11c). There are, however, many drawings of physically realizable objects that may be incorrectly labelled, as in Figure 11e. There are also physically unrealizable objects for which consistent interpretations can be found. Thus, the Huffman-Clowes catalogs do not capture the entire physical and three-dimensional meaning of the line drawing. This is a fundamental problem with a syntactic approach like cataloging.

Waltz (1972) expands the classification of lines to eleven types, including cracks and shadow edges, as well as illumination information for the surfaces on either side. The resulting vertex catalog contained thousands of entries. To avoid combinatorial explosion in the search for consistent interpretations, Waltz used a pseudo-parallel local filtering paradigm. It considered pairs of vertices connected by a common line, and eliminated any vertex interpretations implying an interpretation of the line not allowed by any possible interpretation of the other vertex. This process was iterated over all vertex pairs until no further elimination occurred. At this point, various alternative interpretations of a vertex were assumed, corresponding to one step of a tree search. After each search step, filtering was repeated to limit branching. Surprisingly, for complex scenes, such as Figure 12, the initial filtering step often converged rapidly to a unique solution, with no search needed. The reason appears to be that although the new classes of line introduced by Waltz increase the potential combinatorics, they also increase the constraints to more than compensate: a vertex resulting from a corner casting a shadow has very few alternative interpretations.

Following in Waltz's footsteps, a number of people have attempted to extend the vertex catalog beyond trihedral solids. Turner (1974) treated a domain involving objects with curved surfaces (as in Figure 13) and developed a catalog with tens of thousands of entries. Kanade (1978) allowed sheets as well as solids, and was able to interpret drawings of Origami (paper-folding) constructions (as in Figure 14) as well as simple indoor scenes with planar surfaces.

These attempts to produce extended catalogs for natural scenes expose a second fundamental problem : so many local combinations of edges, surfaces, illumination, and so forth, are possible, that an exhaustive enumeration of scene fragments is almost as impractical as exhaustive enumeration of objects. To control combinatorics, it is necessary to use an even lower level of description - one that makes explicit the physical characteristics of surfaces and their boundaries.

Mackworth's approach to interpreting line drawings (Mackworth 1973) was a step in this direction for blocks-world scenes. He attempted to interpret lines and regions, rather than vertices. Lines had only two basic interpretations: Connect, when the two surfaces visible on either side of the line intersect to form the corresponding solid edge (+ or - Huffman label), and Occluding, when one of the surfaces forming the edge is facing away from the viewer (> Huffman label). Surfaces were represented explicitly by their plane orientations. Each Connect line constrains the relative orientations of the surfaces on either side; if the orientation of one is known, the other surface is hinged to it, and thus has one degree of freedom. Mackworth assumed arbitrary orientation for the first surface considered. He then considered a connected surface and assigned it an arbitrary orientation consistent with the hinge constraint. A third surface meeting the first two is completely constrained. By considering the Connect edges in turn, Mackworth was able to consistently assign surface orientations making the fewest arbitrary assumptions. The remaining problem is to decide which edges are Connect. Mackworth's approach was to make assumptions and then attempt to find surface orientations. If too many lines are assumed to be Connect, the constraints will be inconsistent and no solution will be found. Thus, Mackworth was able to search for all solutions, beginning with those that were most Connected.

The use of explicit surface orientation enabled Mackworth to reject certain interpretations with impossible geometry which were accepted by Huffman, Clowes and Waltz. Since he did not, however, make explicit use of distance information, some geometrically impossible interpretations were still accepted. Geometry is not sufficient by itself to uniquely interpret a line drawing. Horn has demonstrated (Horn 1977) that photometric information can be combined with geometry to provide stronger constraints: for example, the orientation of three surfaces forming a corner may be uniquely determined from the resulting angles of the junction and the relative

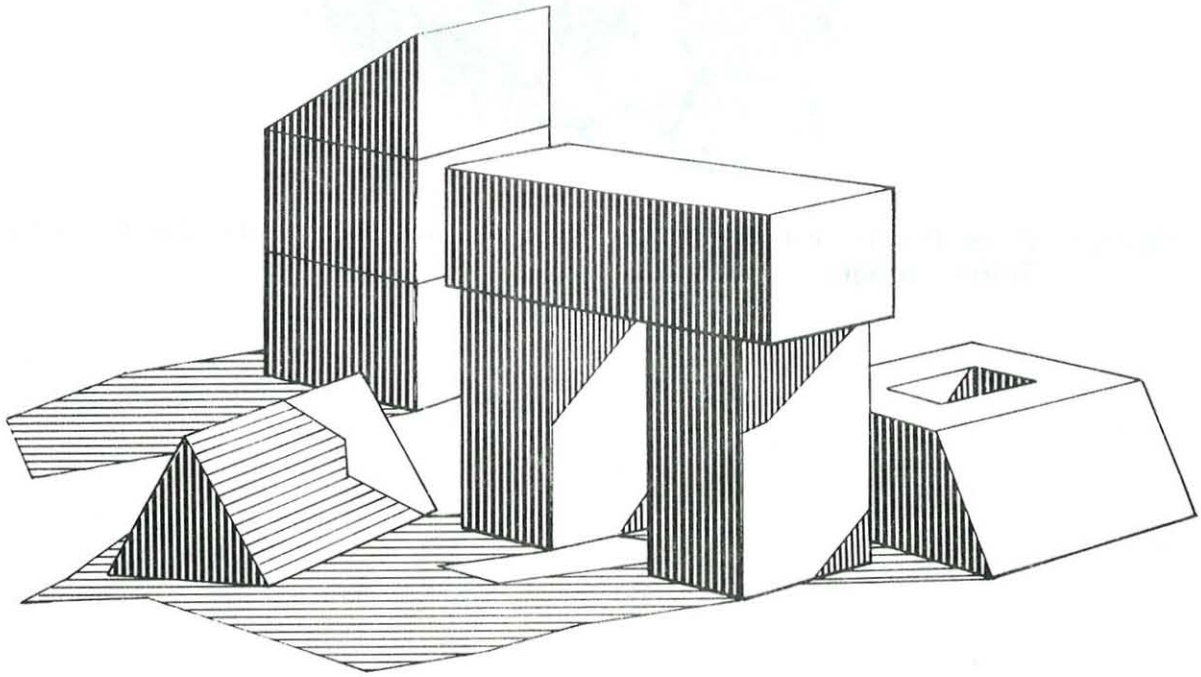


Figure 12 A line drawing correctly interpreted by Waltz's program.

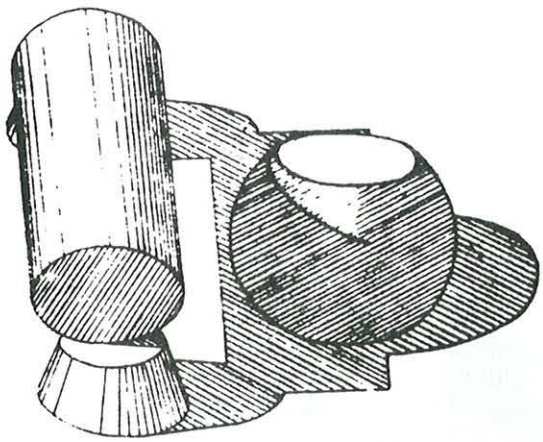


Figure 13 A line drawing of a scene involving objects with curved surfaces labelled by Turner's program.

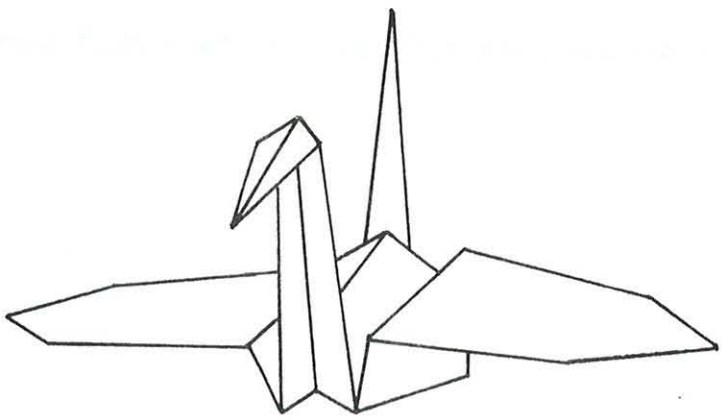


Figure 14 A drawing of an origami object interpretable with Kanade's catalog.

brightnesses of the surrounding regions. Moreover, the physical nature of an edge may often be determinable directly from its brightness profile in a greyscale image (e.g. convex edges often give rise to highlights).

The progression of work, from Roberts to Mackworth and Horn, provides a strong foundation for interpreting images of arbitrary blocks-world scenes. The key ideas are:

- A vision system should be structured as a hierarchy of representations corresponding to a sequence of processing steps that transform a grey level image into a symbolic description of the scene in terms of objects, their positions, orientations and dimensions and their interrelationships.

- Representations and processing should be based upon physical characteristics of the scene and the picture-taking process.

- Processing at each level involves initial interpretation of fragments from local evidence and resolution of ambiguity by eliminating globally inconsistent interpretations.

- Processing cannot proceed in a strictly bottom-up fashion from level to level, but must be guided by what makes sense at higher levels.

- The gap between levels must, however, be sufficiently small to be bridged reliably.

It is believed that the various techniques developed for dealing with the blocks world could be integrated, in accordance with these ideas, into a complete, highly competent vision system. So far, however, such a system has not actually been built.

Natural Scenes

The blocks world illustrates many basic aspects of visual perception. Natural scenes, however, are much more complex than blocks-world scenes, and while the general concepts listed above presumably carry over, specific techniques and representations do not.

In natural scenes, we encounter a potentially infinite range of objects, with very diverse forms, appearances and functions.

The geometric structure of objects usually cannot be modelled by polyhedral prototypes. Objects can be curved, articulated, flexible, even liquid, and their detailed structure can be extremely complex. It is by no means clear how to adequately model the structure of a tree, a crumpled piece of newspaper, a tangle of string or a shaggy dog.

Moreover, there are fundamental constraints that determine, at least at a functional level, the computational process by which knowledge is applied, and the form of the representations. A vision system is naturally structured as a succession of levels of representation, each of which makes explicit a particular aspect of the scene (Feldman et al. 1969, Barrow and Tenenbaum 1975). The initial levels are constrained by what it is possible to compute directly from the image, while higher levels are dictated by the information required to support the ultimate goals. In between, the order of representations and processing is constrained by what is available at preceding levels and by what is required by succeeding ones.

Architecture

Figure 17 outlines what is currently felt to be a plausible computational architecture for a general-purpose vision system capable of high performance in natural scenes.

The sensor encodes the physical characteristics of the scene into a two-dimensional array of brightness values, an input image, which is the initial level of representation in the system. Known geometric or photometric distortions introduced by the sensor may be removed, in a straightforward way, to produce a corrected image that facilitates subsequent processing.

Information in the image is manifested primarily through spatial or temporal intensity changes (and their location in the image) which correspond to changes in some physical characteristic of the scene. The next step is thus to detect these changes and represent them as two-dimensional arrays of feature descriptors: for example, edges might be described by their orientation, width and contrast. These elementary features suffice for simple alerting tasks, and statistics on them provide a basis for elementary discrimination.

Local patterns of image features yield information about the three-dimensional structure of the scene, in the form of texture and shading gradients, local occlusion cues (e.g. line endings), local contour shape, and so forth. From these can be recovered arrays of surface orientation, distance, albedo, and other intrinsic characteristics of the surface element visible at each point in the image (Figure 18).

Simple iconic grouping processes operating on the arrays of intrinsic characteristics can recover regions of homogeneous properties corresponding to three-dimensional surfaces. The notion of a surface is a symbolic abstraction, and this the level at which the transition from iconic representation to symbolic representation can naturally occur.

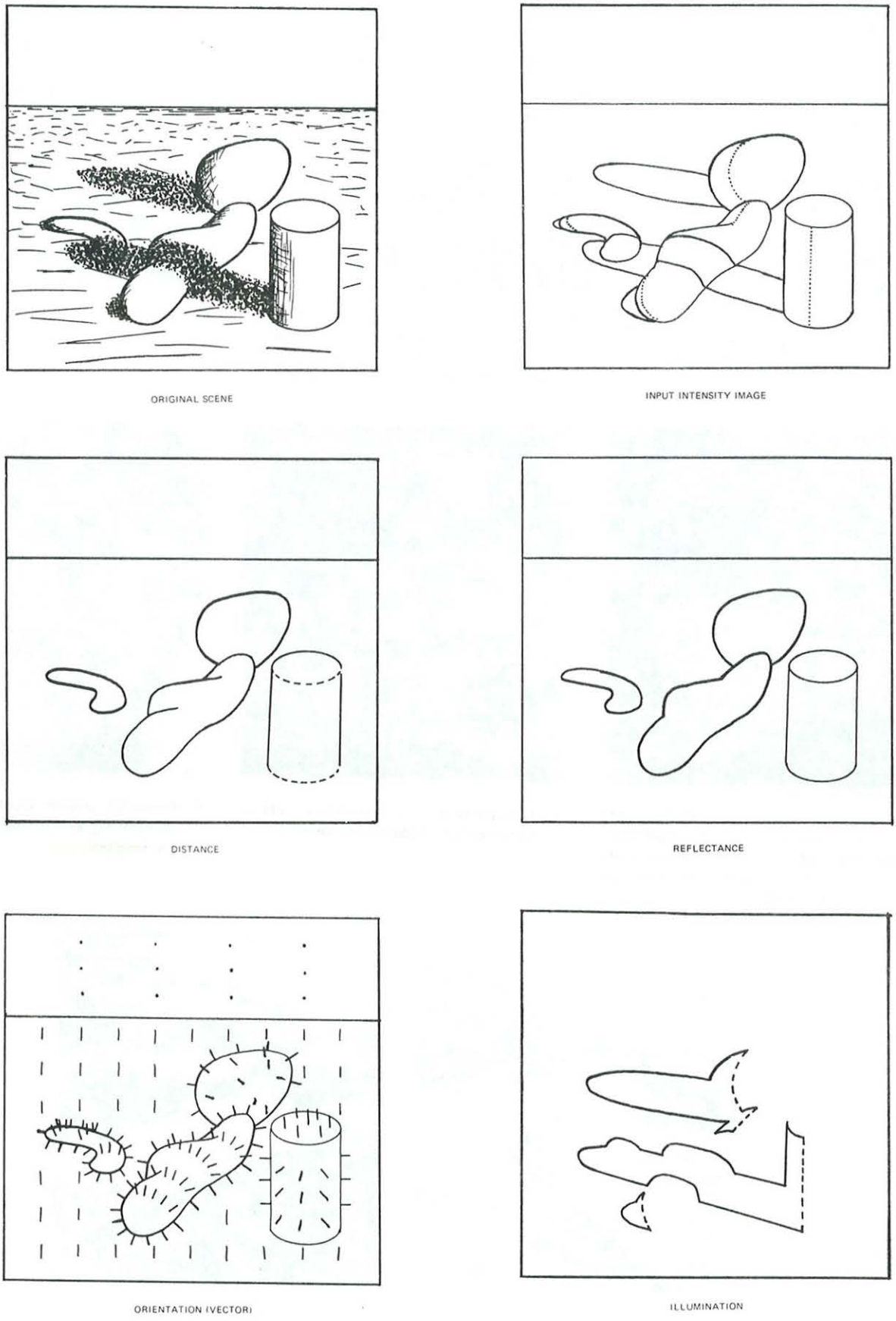
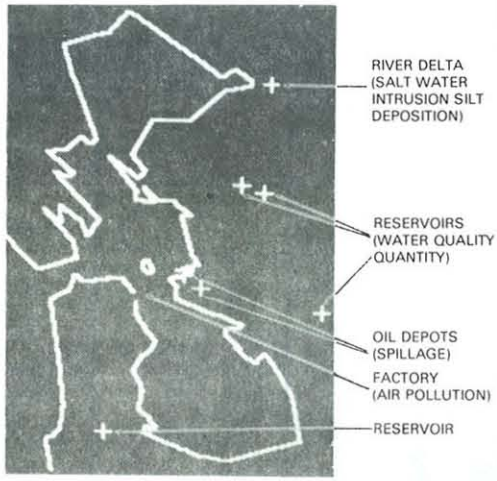
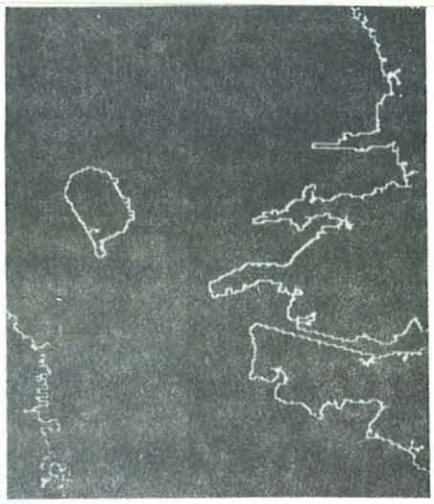


Figure 18 A set of intrinsic images derived from a single intensity image.

a) High altitude vertical mapping photograph of the San Francisco Bay area (Taken from a U-2 at 45,000 feet)

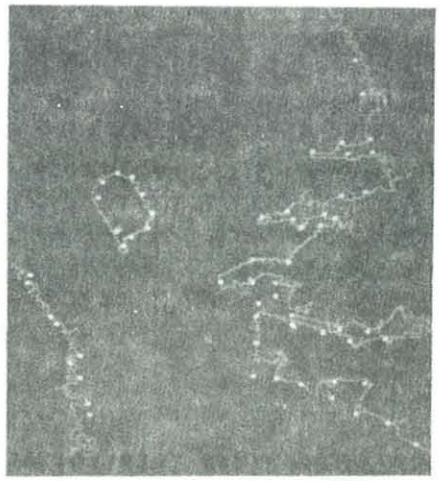


b) Computer display of a simple map data base for the San Francisco Bay area showing major landmark (coastline) and representative monitoring sites (crosses)

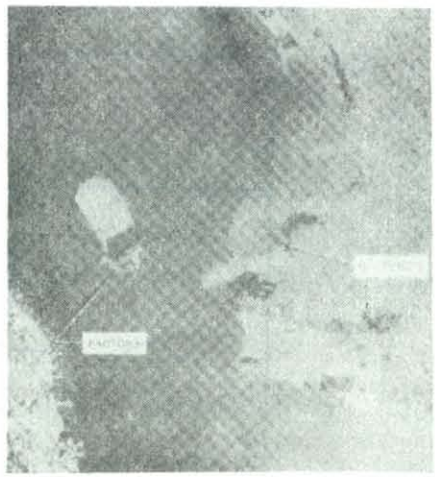


c) Coastline extracted by boundary follower.

d) Predicted image coordinates of coastline (based on navigational estimates of camera location and orientation) superimposed on extracted boundary.



e) Predicted coordinates after optimization of camera parameter.



f) Predicted image locations of visible monitoring sites based on optimized parameters.

g) Predicted locations of visible monitoring sites in an oblique view looking west from Alameda.

h) Predicted locations of visible monitoring sites in a high altitude oblique view looking east from the Pacific ocean.

Figure 19 Establishing correspondence between an image and a map using parametric correspondence.

Symbolic grouping processes organize surfaces into sets corresponding to distinct bodies (as suggested by Guzman). The symbolic representation of bodies may include volumetric primitives that make explicit the space occupied by an object and not just its visible surfaces.

Bodies so described are then recognized as known objects, or new ones, and symbolic representations constructed to describe them. Object representations include information about three-dimensional location and orientation, and pointers to generic descriptions, as well as characteristics that may not be directly visible.

At this point, the boundary between Vision and Cognition becomes blurred and our ideas on architecture become less certain. Presumably, objects are grouped into yet higher-level configurations, comprising scenes and events, and arbitrarily complex reasoning processes may be required.

We can be reasonably certain, on computational grounds, that each of these levels of description should be present in a visual system. There can be other levels, one possibility being a two-dimensional organization of image features into connected boundaries or closed regions, prior to recovery of physical characteristics. Although we have described the processing in a strictly bottom-upward, data-driven sequence, corresponding to the primary direction of information flow, we know that some information must also flow top-downwards (goal-driven) to ensure meaningful results at higher levels.

Current Systems

In this section examples are presented of some representative state-of-the-art computer vision systems that embody principles set forth previously. There is, at present, no single implemented system that encompasses all of the levels of representation included in the conceptual design. However, a number of systems have been built, each exploiting some of the basic principles to achieve useful performance in a limited context. Moreover, some ambitious attempts at general-purpose vision systems are currently in progress.

Map-Guided Interpretation of Aerial Imagery

An important application of aerial imagery involves the continuous long-term monitoring of designated ground sites. Examples include monitoring particular industrial plants for pollution, oil storage facilities for spillage, and reservoirs for water quality. Ideally, an automated system should be capable of extracting updated information as new imagery arrives for distribution to interested users.

A major problem in automating such tasks is locating the designated sites in sensed imagery that may be taken from arbitrary viewpoints. Once the image locations of sites are known, many monitoring tasks are reduced to straightforward detection or classification problems (e.g. classifying the multi-spectral signature of a pixel located in a stream beside a factory to detect pollution). Ground locations have conventionally been determined by warping the current sensed image into correspondence with a reference image, based on a large number of local correlations (Bernstein 1976). This process is, however, computationally expensive and limited to cases in which the reference and sensed images were obtained under similar viewing conditions.

The Hawkeye system, developed at SRI (Barrow 1977, Tenenbaum et al. 1980), overcomes these limitations by exploiting two models: a model of the scene in the form of a symbolic map, and a geometric model of the camera. The map contains three-dimensional coordinates of sites to be monitored as well as landmarks (roads, coastlines, and so forth). The geometric correspondence between this map and the sensed image is established by calibrating the camera model on the landmarks. The camera model is then used to predict the precise image locations of the sites in question, and task-specific operators are applied at these locations.

A novel calibration process, called Parametric Correspondence (Barrow et at. 1977) was developed for map matching. It is illustrated in Figures 19a-f. Initial estimates of camera location and orientation are obtained on the basis of navigational data. The camera model is then used to predict the location of landmarks in the image for this assumed viewpoint. The camera parameters (i.e. the assumed viewpoint) are then adjusted to make the predicted locations optimally match the locations of corresponding features extracted from the image. Figures 19g and 19h provide two additional examples, illustrating the ability of the calibration process to place the map into correspondence with imagery taken from arbitrary viewpoints. This flexibility is an excellent example of the power gained by using the right models: no correlation warping process can handle such diverse viewpoints.

Having placed an image into correspondence with a map, many basic monitoring tasks can be automated with straightforward techniques. In Figure 19f, for example, the pixels located in reservoirs can be tested for water quality, the pixels located in shipping channels beside oil depots for evidence of spillage, the pixel located at the industrial plant for evidence of particulates, and the pixel located at the Sacramento River delta for evidence of salt-water intrusion. All of these applications are well-known in the remote sensing literature (Greeves, Anson and Landen 1975). However, given the world model represented by the map, and some simple task-specific models, a number of previously intractable tasks can be handled easily.

Water levels in reservoirs can be monitored by extracting the outline of the reservoir in the image and determining its location with respect to elevation contours from a registered topographic map. The image coordinates of selected points on the reservoir boundary are determined to sub-pixel precision by analyzing the gradient of intensity along a line in the image perpendicular to the contours of each point. Elevations are determined at a number of points and are averaged together to compensate for statistical uncertainties in estimating boundary points precisely.

Object detection, mensuration and counting can be facilitated by using the map to constrain where to look and what to look for. For example, the number of box-cars in a rail yard can be counted by looking along predicted paths of railroad track in an image for changes in brightness and dark transverse lines (which signify the ends of cars). Hypothesized ends are interpreted in the context of knowledge about trains (e.g. standard car lengths and allowed inter-car gap widths) and about the characteristics of empty track to prune artifacts and improve the overall reliability of interpretation. Similarly, ship monitoring is accomplished by analyzing intensity patterns alongside predicted berth locations in a harbour to distinguish ships from water; water characteristically has a low density of edges.

The Hawkeye system illustrates that higher-level knowledge, such as that provided from a map, can compensate for primitive low-level descriptions in a performance system. Other advantages include the ability to focus processing on the relevant portions of the image, sharply reducing computational costs, and to use processing methods that can exploit knowledge of what to look for at each site. However, for many applications, such as map-making, such detailed scene knowledge is unavailable. The following system addresses such a problem.

Detection of Linear Features

Detecting linear features is an important requirement in many vision applications. In cartography, for example, roads, rivers and railroads all appear in low-resolution aerial imagery as lines with no internal structure (Figures 20a, c and e).

Two fundamental problems must be addressed in a competent line extraction program. First, the local appearance (width, contrast) of linear features can vary significantly in a single image, not to mention all possible images. Thus many alternative models of road appearance are needed. Second, the perception of a linear feature requires global models of continuity to bridge local areas where appearance is corrupted. (See, for example, the road going through the forest in the upper right corner of Figure 20a). Not surprisingly, early attempts at road-finding, which relied on a simple linear template to estimate the likelihood of a road being present at each point in the image, performed poorly.

A road extraction program recently developed at SRI (Fischler, Tenenbaum and Wolf 1979) can overcome these difficulties by using many alternative models of road appearance to estimate local likelihood. These models can include linear and non-linear templates of various sizes and orientations, as well as procedural models that test for such conditions as brightness, contrast, width, curvature, parallel edges, and so forth. The results of these local decisions are integrated by a global optimization model which evaluates the cumulative likelihood for all alternative paths in the image, using dynamic programming.

A fundamental problem with model-based approaches is how information from incommensurate models should be combined in reaching a global decision. The approach taken in this system was to partition the local models into two classes based on their error characteristics: type I models that will almost never incorrectly classify artifacts as instances of the structure for which they are searching, but may often miss correct instances; and type II models that accurately measure relevant parameters of all true instances, but may falsely detect and incorrectly measure non-instances. Coherent responses from type I models are superimposed to obtain a reliable sketch that is filled in using type II models. The type II models are applied independently in the vicinity of the sketch, and at each point, the model that is most confident of its output is accepted.

The program has been tested extensively on over a hundred image fragments and has performed at near-human levels. Figures 20b, d and f show a sampling of its results.

Model-Based Photo Interpretation

The ACRONYM system at Stanford University (Brocks, Greiner and Binford 1979) uses multiple levels of representation, and a combination of data-driven and goal-directed control to recognize objects in aerial imagery (see Figure 21). The initial level extracts and links intensity discontinuities in an image (see Figure 22) into edge fragments (Figure 23). Stereo data, when available, allows edge linking to be done in three dimensions, reducing ambiguities in two ways: edges on different surfaces that appear aligned from a particular viewpoint are kept distinct; edges on or beneath the support plane can be eliminated on the basis of height to reduce clutter and hence the combinatorics of linking (Figure 24).

Pairs of parallel edge fragments with opposing contrasts are grouped into an intermediate representation, called ribbons, that correspond to projections of objects modelled as generalized cylinders. Quasi-projectively-invariant features of ribbons (such as whether their axis of symmetry is straight or curved, whether they are elongated of squat, and how they connect to other ribbons) are used to hypothesize possible object prototypes and projections (viewpoints). The hypothesized model is then transformed

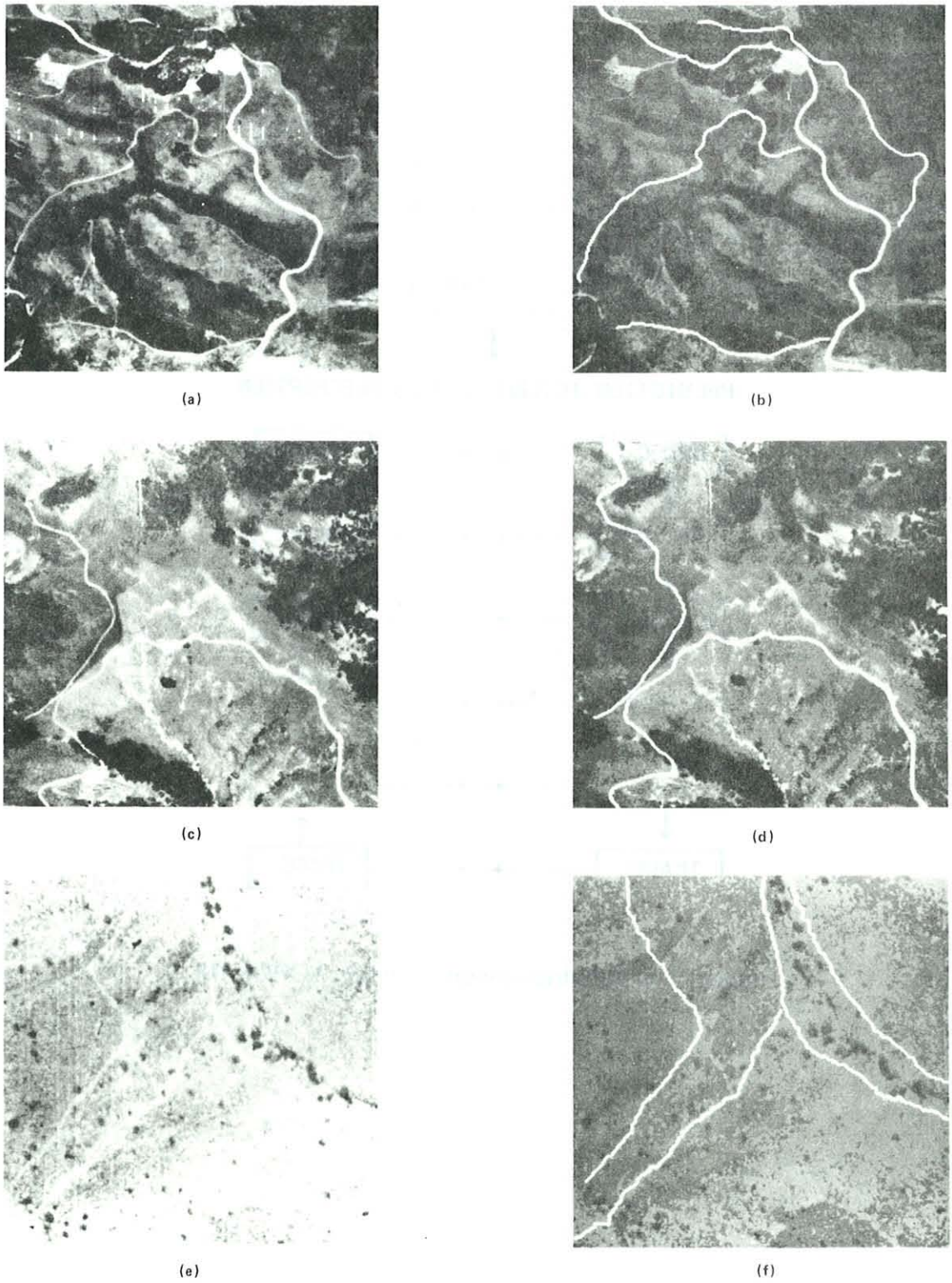


Figure 20 Delineation of roads in low-resolution aerial imagery.

a, c, e) Original images
b, d, f) Roads traced automatically

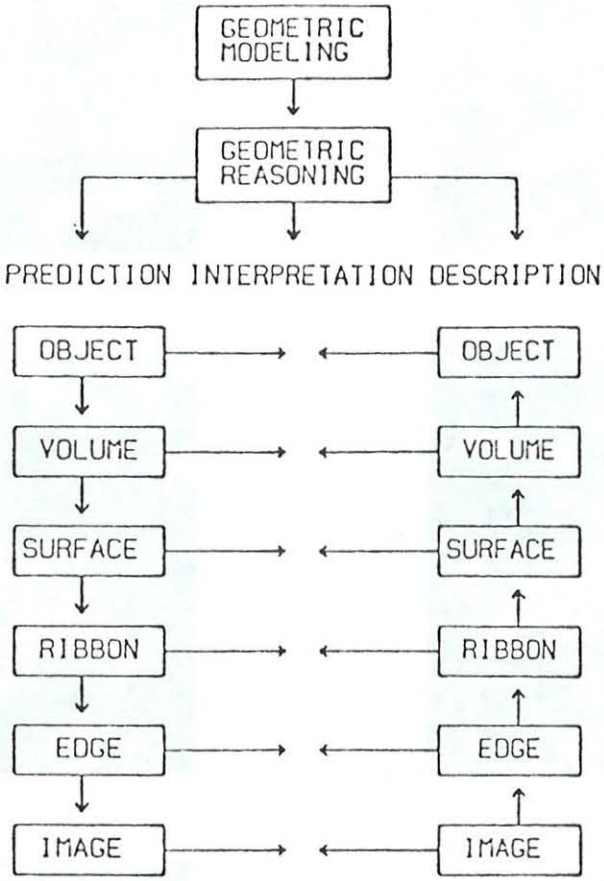


Figure 21 Geometric reasoning within ACRONYM.

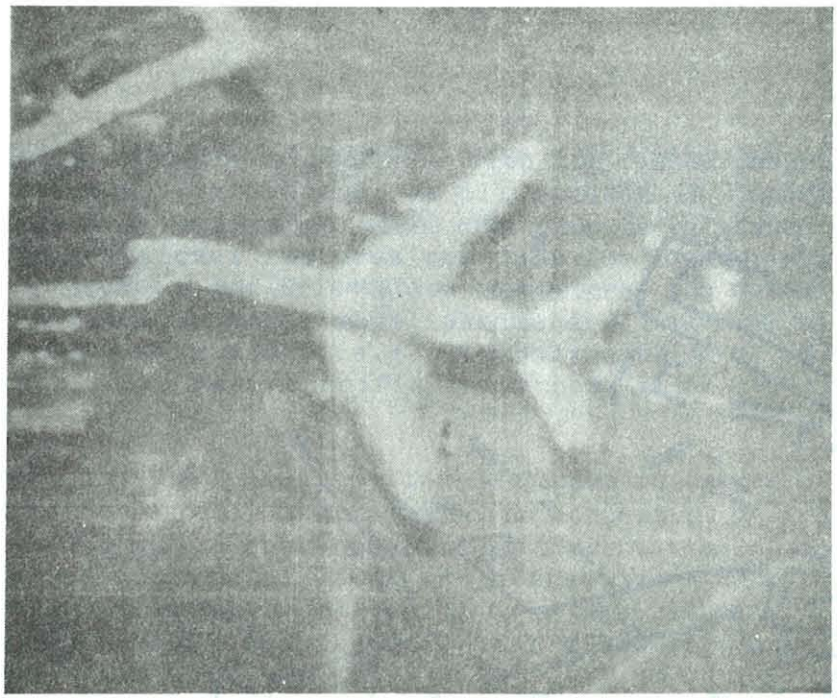


Figure 22 LH frame from stereo pair of aerial photographs

Figure 23 Detected edge fragments.

Objects are modelled, as in ACRONYM, by a symbolic description tied to geometric primitives (i.e. a schema). Figure 28 shows the detail present in these models. Object recognition is accomplished using an MSYS-style (Barrow and Tenenbaum 1976) matching procedure, based on shape attributes of the prototypes in conjunction with spectral features, texture, size (estimated from perspective cues), and the context provided by the schema.

The VISIONS system has been under development for several years and an impressive amount of knowledge has been collected and formalized. To date, experimentation has been limited, however, to top-down use of schema to refine pictorial segmentation and interpretation in a manner closely analogous to IGS (Tenenbaum and Barrow 1977). The principal differrence is that IGS used descriptions of scenes from a particular viewpoint, whereas VISIONS is able to generate such descriptions for a particular viewpoint from its three-dimensional object models.

DISCUSSION

Dr. Grossman asked whether Dr. Barrow could supply "numbers" so that quantitative comparisons might be made. In particular he wondered whether it took weeks or months to program a vision system to recognize scenes, and how reliable the systems described were.

Dr. Barrow replied that the design of a sophisticated, reliable, general-purpose vision system was still the subject of research. The implementation of the more advanced experimental systems has involved several man-years of effort. Once built, however, it took a comparatively short time to give such a system the capability of recognizing a new object. As an example, he cited a commerical company - an SRI spin-off - marketing the vision module which could be "trained" in minutes on a number of objects.

Professor Randell pursued the question of doing a job reliably. **Dr. Barrow** could give no precise figures for experimental general-purpose vision systems. Simpler systems such as a vision module in constrained situations could be "pretty reliable". The overall reliability of an assembly task, for example, could be made much higher by using a simple visual check that a bolt had been actually inserted, and taking corrective action if necessary.

Dr. Larcombe asked about the size of the terrain maps described in the lecture. It was stated that they were of about 2000 x 2000 points with perhaps 16 bits per point. Larger data bases were anticipated with the image on disk paged into memory as required. In response to a question about programming languages used, **Dr. Barrow** said that the Hawkeye system used Interlisp, Sail and some assembly code on a DEC 10 computer. The system occupied 13 address spaces (of 256K words), though many of them were only partially full, so some compression was possible.

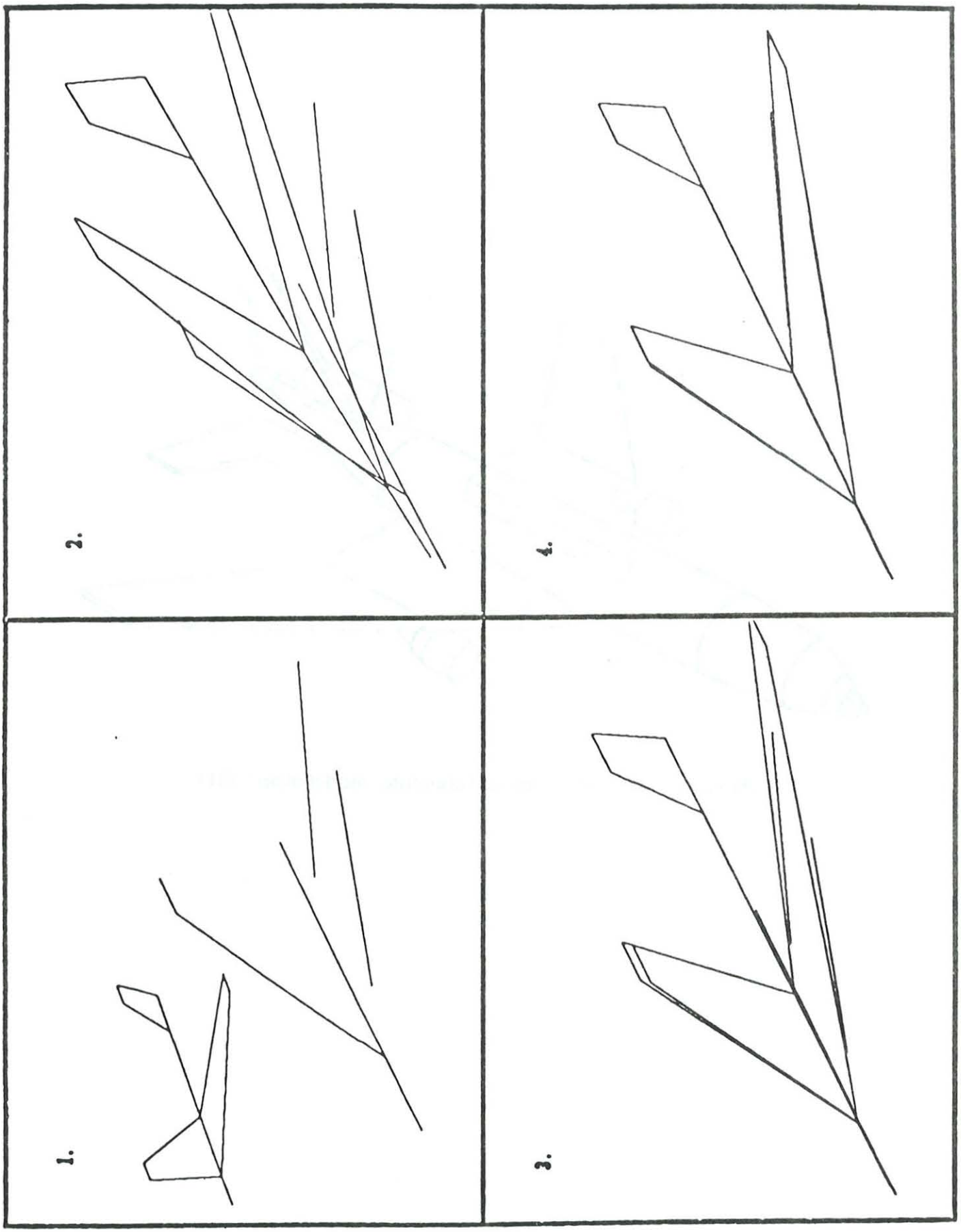


Figure 25 Matching a simple 3-D model to lines found in an image.

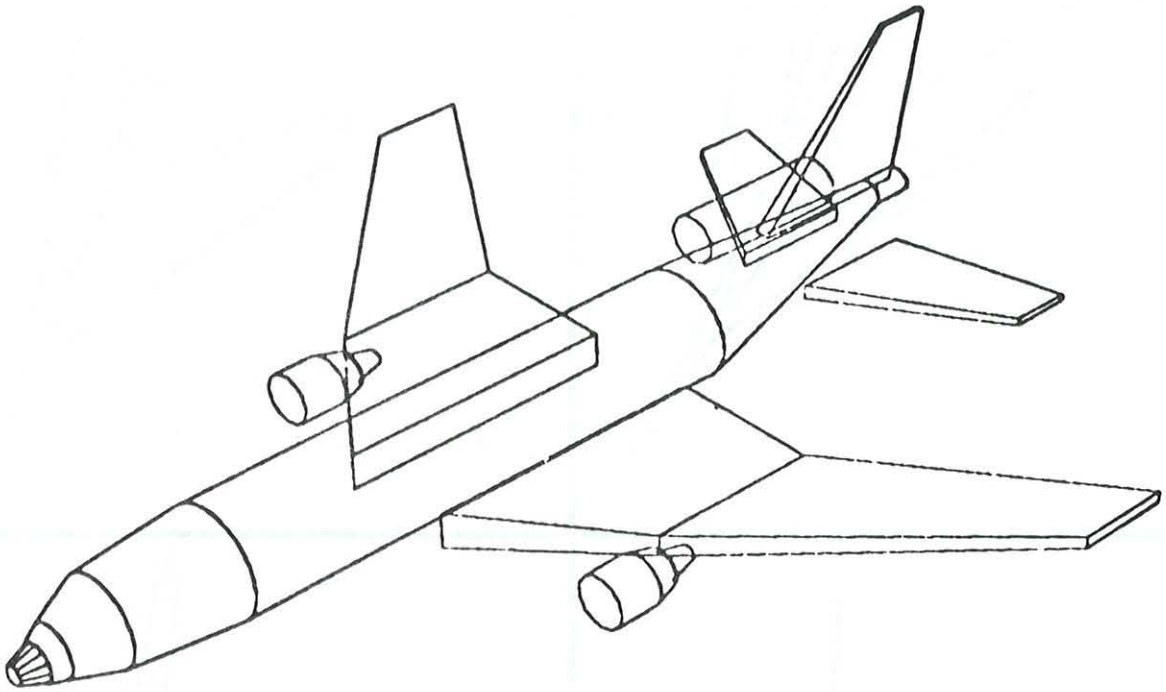


Figure 26 Three-dimensional computer model of an L1011.

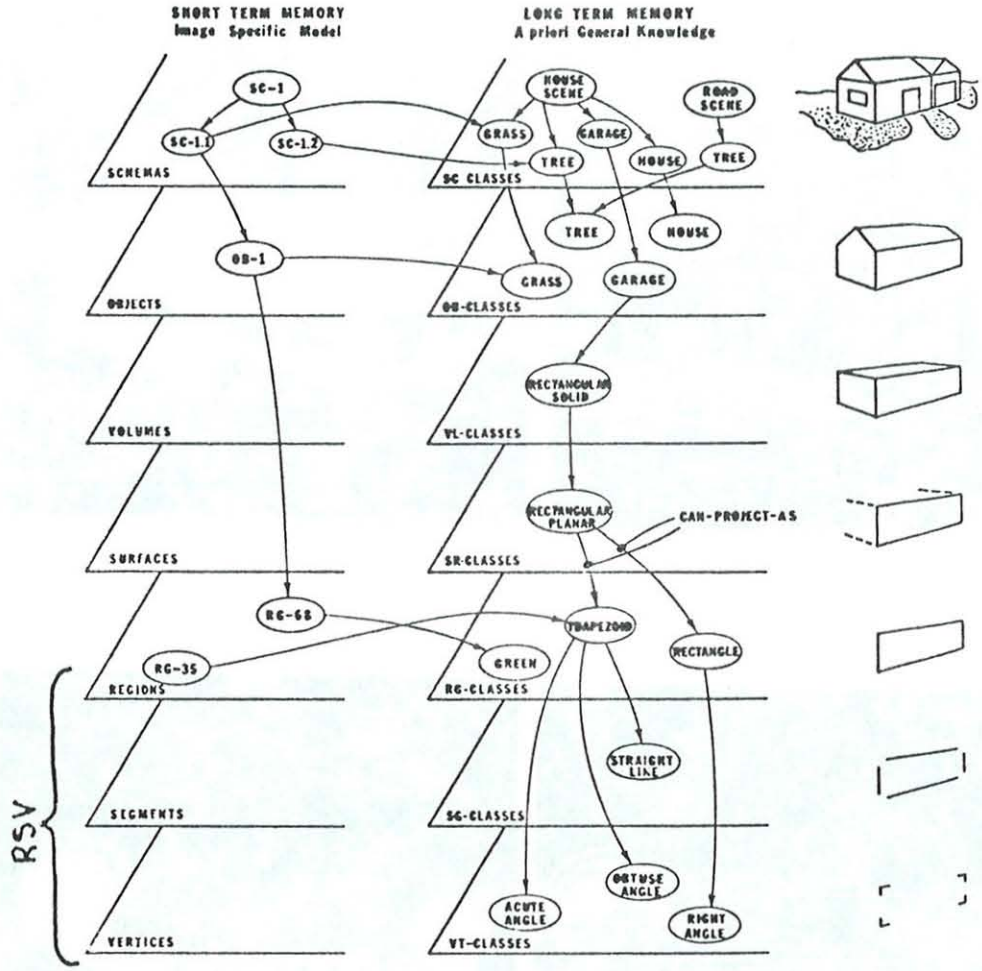


Figure 27 Hierarchical decomposition of long-term memory (LTM) and its relationship to short-term memory (STM) in the VISIONS system. LTM contains the stored knowledge to which the system has access. An interpretation of an image is viewed as a set of instantiations in STM of nodes in LTM.

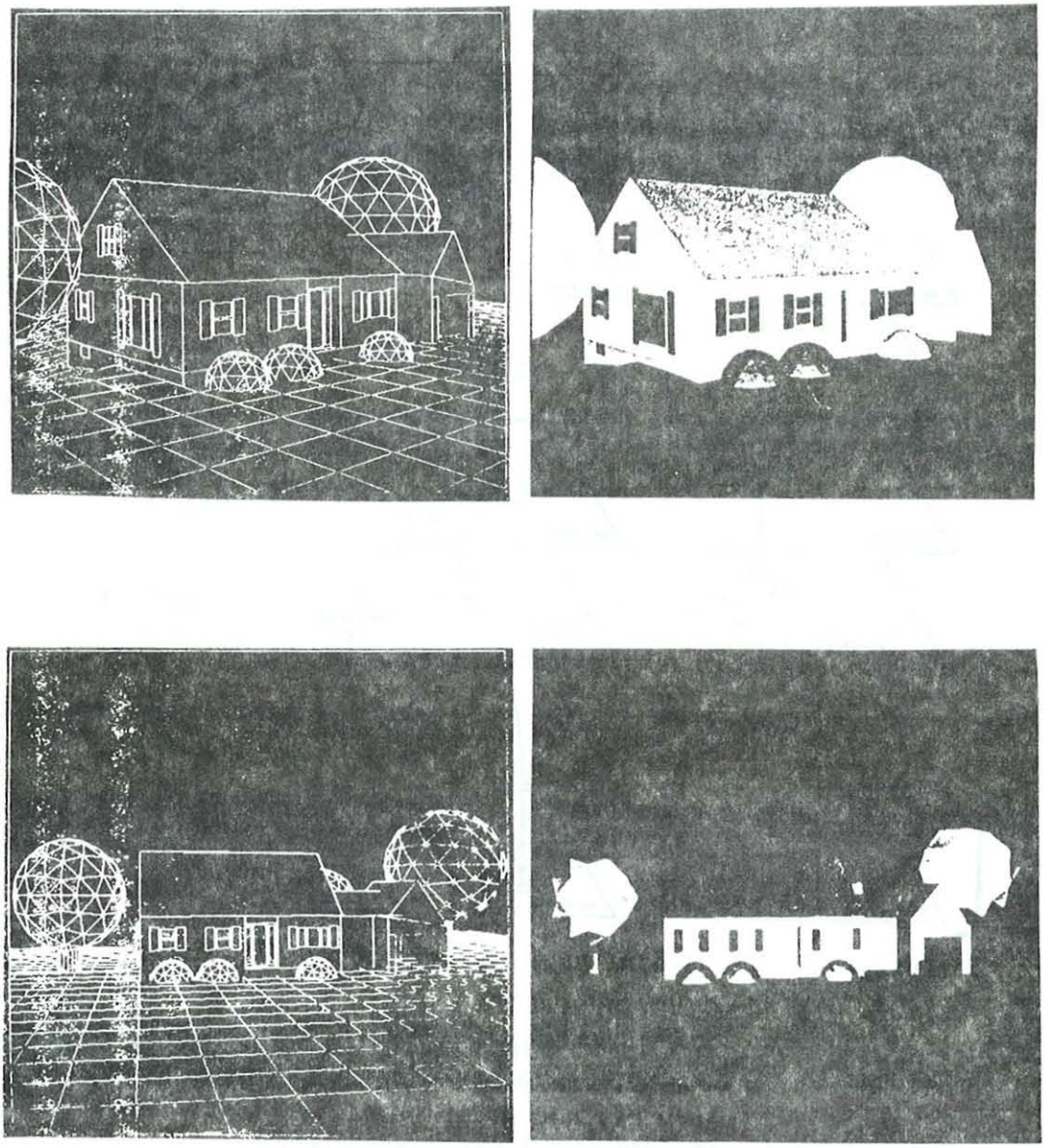


Figure 28

Wire frame and surface representations of a model of the house image seen from two points of view. The current 3D house scene schema is actually an abstract representation of the approximate relative spatial locations of the entities in these images. The components (volumes, surfaces, straight line segments) are actually represented by a position in space and a radius associated with a decreasing likelihood of the component appearing at that location.

REFERENCES

G.J. Agin and T.O. Binford (1976). "Computer Description of Curved Objects", IEEE Transactions on Computers, April 1976, pp. 439-449.

R. Bajcsy (1973). "Computer Description of Textured Surfaces" Proc. Third Intl. Joint Conference on Artificial Intelligence, Stanford University, Stanford, CA, August 1973, pp. 572-579.

R. Bajcsy and L. Lieberman (1976). "Texture Gradient as a Depth Cue", Computer Graphics and Image Processing, Vol. 5, 1976, pp. 52-67.

H.G. Barrow (1977). "Interactive Aids for Cartography and Photo Interpretation", Semiannual Technical Report, SRI International, Menlo Park, CA, December 1977, Contract DAAG29-76-C-0057, SRI Project 5300.

H.G. Barrow and R.J. Popplestone (1971). "Relational Descriptions in Picture Processing", in Machine Intelligence 6, (1971), B. Meltzer and D. Michie, eds., Edinburgh University Press, Edinburgh, Scotland, 1971, pp. 377-396.

H.G. Barrow and J.M. Tenenbaum (1975). "Representation and Use of Knowledge in Vision", Technical Note 108, SRI International, Menlo Park, CA, July 1975.

H.G. Barrow and J.M. Tenenbaum (1976). "MSYS: A System for Reasoning about Scenes", Technical Note 121, SRI International, Menlo Park, CA, April 1976.

H.G. Barrow and J.M. Tenenbaum (1978). "Recovering Intrinsic Scene Characteristics from Images", in Computer Vision Systems, A. Hanson and E. Riseman, eds., Academic Press, New York, New York, 1978, pp. 3-26.

H.G. Barrow et al. (1977). "Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching", Proc. Fifth Intl. Joint Conference on Artificial Intelligence, M.I.T., Cambridge, Mass., August 1977, pp. 659-663.

R. Bernstein (1976). "Digital Image Processing of Earth Observation Sensor Data", IBM Journal of Research and Development, Vol. 20, No. 1, January 1976.

C. Brice and C. Fennema (1970). "Scene Analysis Using Regions", Artificial Intelligence, Vol. 1, No. 3, 1970, pp. 205-226.

R.A. Brooks, R. Greiner and T.O. Binford (1979). "The ACRONYM Model-Based Vision System", Proc. of the Sixth Intl. Joint Conference on Artificial Intelligence, Tokyo, Japan, August 1979, pp. 105-113.

M.B. Clowes (1971). "On Seeing Things", Artificial Intelligence, Vol. 2, No. 1, 1971, pp. 79-112.

G. Falk (1972). "Interpretation of Imperfect Line Data as a Three-Dimensional Scene", Artificial Intelligence, Vol. 4, No. 2, 1972, pp. 101-144.

J. Feldman et al. (1969). "Stanford Hand-Eye Project", Proc. International Joint Conference on Artificial Intelligence, Washington, D.C., 1969, pp. 521-526.

M.A. Fischler, J.M. Tenenbaum and H.C. Wolf (1979). "Detection of Roads and Linear Structures in Low-Resolution Aerial Imagery Using a Multisource Knowledge Integration Technique", Technical Note 200, SRI International, Menlo Park, CA, December 1979. (To be published in : Computer Graphics and Image Processing, in August 1980.)

R.G. Greeves, A. Anson and D. Landen (1975). Manual of Remote Sensing, American Society of Photogrammetry, Falls Church, Virginia 22046, 1975.

A. Guzman (1968). "Computer Recognition of Three-Dimensional Objects in a Visual Scene", MAC-TR-59 (Thesis), Project MAC, M.I.T., Cambridge, Mass. 1968.

A.R. Hanson and E.M. Riseman (1978). "VISIONS: A Computer System for Interpreting Scenes", in Computer Vision Systems, A.R. Hanson and E.M. Riseman, eds., Academic Press, New York, New York, 1978, pp. 303-333.

R.M. Haralick (1979). "Statistical and Structural Approaches to Texture", Proc. IEEE, Vol. 67, May 1979, pp. 786-804.

B.K.P. Horn (1975). "Obtaining Shape from Shading Information", in The Psychology of Computer Vision, P.H. Winston, ed., McGraw-Hill, New York, New York, 1975.

B.K.P. Horn (1977). "Understanding Image Intensities", Artificial Intelligence, Vol. 8, No. 2, 1977, pp. 201-231.

B.K.P. Horn (1979). "SEQUINS and QUILLS - Representations for Surface Topography", AI Memo 536, M.I.T., Cambridge, Mass., May 1979.

D.A. Huffman (1971). "Impossible Objects as Nonsense Sentences", in Machine Intelligence 6, B. Meltzer and D. Michie, eds., Edinburgh University Press, Edinburgh, Scotland, 1971, pp. 295-323.

T. Kanade (1978). "A Theory of Origami World", Tech. Report CMU-CS-78-144, Carnegie-Mellon University, Pittsburgh, PA, September 1978.

K.I. Laws (1980). "Textured Image Segmentation", Ph.D. dissertation, University of Southern California, January 1980.

G.G. Lendaris and G.L. Stanley (1970). "Diffraction-Pattern Sampling for Automatic Pattern Recognition", Proc. IEEE, Vol. 58, February 1970, pp. 198-216.

A.K. Mackworth (1973). "Interpreting Pictures of Polyhedral Scenes", Artificial Intelligence, Vol. 4, 1973, pp. 121-138.

J.T. Maleson, C.M. Brown and J.A. Feldman (1977). "Understanding Natural Texture", Proc. Image Understanding Workshop, DARPA, October 1977, pp. 19-27.

D. Marr (1976). "Early Processing of Visual Information", AI Memo 340, M.I.T., Cambridge, Mass., 1976.

D. Marr (1978). "Representing Visual Information", in Computer Vision Systems, A. Hanson and E.M. Riseman, eds., Academic Press, New York, New York, 1978, pp. 61-80.

D. Marr and H.K. Nishihara (1977). "Representation and Recognition of the Spatial Organization of Three-Dimensional Shapes", Proc. Roy. Soc. B. Vol. 200, 1977, pp. 269-294.

L.G. Roberts (1965). "Machine Perception of Three-Dimensional Solids", in Optical and Electro-Optical Information Processing, J.T. Tippett et al., eds., M.I.T. Press, Cambridge, Mass., 1965.

A. Rosenfeld and A.C. Kak (1976). Digital Picture Processing, Academic Press, New York, New York, 1976.

B.R. Schatz (1977). "The Computation of Immediate Texture Discrimination", AI Memo 426, M.I.T., Cambridge, Mass., August 1977.

Y. Shirai (1973). "A Context Sensitive Line Finder for Recognition of Polyhedra", Artificial Intelligence, Vol. 4, No. 2, 1973, pp. 95-119.

K.A. Stevens (1979). "Surface Perception from Local Analysis of Texture and Contour", Ph.D. dissertation, M.I.T., Cambridge, Mass., February 1979.

J.M. Tenenbaum, H.G. Barrow and S.A. Weyl (1975). "Research in Interactive Scene Analysis", Final Report, Stanford Research Institute, Menlo Park, CA, 1975, Contract NASW-2086, SRI Project 8721.

J.M. Tenenbaum et al. (1980). "Map-Guided Interpretation of Remotely Sensed Imagery", Proc. National Computer Conference, Anaheim, CA, June 1980, pp. 391-408.

K.J. Turner (1974). "Computer Perception of Curved Objects Using a Television Camera", Ph.D. dissertation, Edinburgh University, November 1974.

D.L. Waltz (1972). "Generating Semantic Descriptions from Drawings of Scenes with Shadows", Tech. Report AI-TR-271, M.I.T., Cambridge, Mass., November 1972.

A.P. Witkin (1980). "Shape from Contour", Ph.D. dissertation, M.I.T., Cambridge, Mass., February 1980.

Y.Y. Yakimovsky and J. Feldman (1973). "A Semantics-Based Decision Theoretic Region Analyzer", Proc. Third Intl. Joint Conference on Artificial Intelligence, Stanford University, Stanford, CA, 1973, pp. 580-588.