Looking at macromolecules: hints from the artificial intelligence of vision

Carl Trindle

Chemistry Department, University of Virginia, Charlottesville, VA 22901 USA

The study of computer vision provides valuable insights that can aid the construction of an expert system intended for molecular modeling. In this paper, I will begin by providing a brief and general description of expert systems. Then I will provide at least a caricature of the construction of an expert system that deals with some of the problems encountered in molecular modeling with computer graphics. Finally, I will make suggestions drawn from the investigation of computer vision that are guiding our development of DOCENT, an expert system intended for application in molecular modeling.

Keywords: computer vision, DOCENT, expert system

Computer graphics plays a key role in molecular modeling[1] and has changed that venerable practice drastically. Modeling, which has been an indispensable part of chemical reasoning for more than a century, has gained a flexibility and power that would astound those who mocked van't Hoff.[2] With training and practice, a user of a molecular graphics workstation can view the invisible as if it fit naturally on a desktop. The model's image can be rotated, magnified and represented as suits the user. Fragments can be manipulated to approach one another and joined to test the ease of fit of a substrate and a binding site. Clues from the visualization of the binding contribute to the design of substrates that bind more effectively and enhance the activity of complex systems.

The human user is a critical component of the modeling process. In fact, a frequent debate concerns what aspects of human perfomance can be captured in computer code. It is obviously desirable to code as much as possible of the work of modeling that is not germane to chemical considerations, and that might distract attention from the researcher's creative problem-solving. So we code details of image projection, the matrix algebra of coordinate-frame rotation, algorithms for removing portions of a scene that nearby objects screen from view, shading techniques and so on.[3] It should not be forgotten that the purpose of the coding is not only to present as much as feasible of the geometric and energetic information within the model, but also to make it easier for the user to operate a complex and powerful system. This includes guaranteeing an easy and encouraging first encounter with the system and assuring rapid progress toward use of the system in problems of chemical significance.

Molecular modeling uses computer graphics as if the human user were an expert to whom many subtle judgments must be deferred. The expert infers three-dimensional (3D) information from a two-dimensional (2D) image, selects axes about which the molecule can be rotated, reorients fragments by distorting the molecule along soft internal modes of torsion and vibration, folds macromolecules to alter their topography and recognizes promising features of the molecular topography that typify hospitable docking sites, where molecules may "fit" together. The human expert can read the implications of displays that represent surfaces by filling or speckling, color-code the electrostatic potential and show the structure of the molecule by backbones or ribbons. All this is easy for the expert but difficult for the novice.

It is important to the novice to achieve a degree of expertise rapidly. One effective way is to be led through some example sessions by an expert user who is available not only for the initial tutorial but for consultation when the novice encounters difficulties. Every vendor of elaborate equipment tries to provide this kind of counsel in the form of manuals, tutorials and telephone assistance. But we all know how difficult it is to find competent advice and guidance. Particularly in primarily visual domains, where all that the expert knows is not easily expressed in words, manuals offer little aid. Apprenticeship is invaluable; the novice gains skill not only by formal (oral) explanation but by imitation of essential but unexplained gestures of the master. If only the workstation itself could incorporate the expertise needed to train and lead the user from first acquaintance to mastery, its value would be greatly enhanced.

The guidance of the master is all the more important when you wish to accomplish tasks that resist rote application of algorithms. Complex tasks — including

research, by definition — invariably require more than algorithms. These tasks call for rules of thumb, decisions taken with incomplete knowledge, even "hunches" that become second nature for an expert but are absolutely mysterious to the novice. These "heuristics" — the word is drawn from a root suggesting "leading toward discovery" — are hard to articulate but seem central to the remarkable performance of experts.

Artificial Intelligence (AI) techniques of programming can help. The use of (near) natural language, the automation of inference and particularly the incorporation of heuristics can all help the novice achieve results that rival the best an expert working without AI assistance could provide.

One strikingly successful branch of AI research deals directly with problem-solving, in a specialized domain, incorporating heuristics as well as algorithms.[4] "Expert systems" are realizations in software of these techniques.

WHAT CAN EXPERT SYSTEMS DO?

Expert systems assume a number of tasks.[5] They deduce chemical structures from spectroscopic data (META-DENDRAL); they analyze requested VAX computer configurations for consistency and generate a parts list and assembly diagram (R1); they diagnose medical disorders from clinical data (MYCIN, PUFF, INTERNIST); and they design experiments in molecular genetics (MOLGEN) and identify promising geological sites (PROSPECTOR). In principle they can be used for subtle tutoring and for planning, monitoring and debugging. In short, they do a variety of tasks that require expert insight and judgment, which we often take to be the product of experience. They have in common a focus on a small data-rich domain of knowledge.

Expert systems (and AI efforts in general) have found the following tasks particularly difficult: natural language processing, image interpretation, and geometric and spatial modeling. Successful expert systems such as MYCIN and PROSPECTOR, and the widely used expert system precursors (Newell's view) MACSYMA and DENDRAL have for the most part bypassed these problems. Currently successful expert systems use "backward chaining" logic on a collection of "production rules"; the precursors used pattern-matching to formulas to achieve symbolic integration (MACSYMA) or generate-and-test methods exhausting all possible chemical structures (DENDRAL). In the precursors the computer program does not necessarily follow the rules or methods preferred by a human expert.

Expert systems can be good servants but bad masters. They fail outside their realm of expertise, sometimes without warning. But used with caution (like other experts), an expert system can be a useful advisor, guide or famulus, just as Wagner was to Faust. In a molecular modeling system using computer graphics, an expert system could adopt the role of tutor, guide and assistant to the novice and could ease the work of even an accomplished user and interpreter of molecular graphics.

WHERE IS THE EXPERTISE?

One of the earliest exercises in natural language processing (still a very difficult problem for AI) was Joseph Weizenbaum's program ELIZA,[6] in which a Rogerian nondirective counselor was caricatured. So long as the user was content to play along with the counselor — that is, to stay in the very small domain where ELIZA's collection of stock phrases made some sense — it was possible to maintain a plausible "conversation." To Weizenbaum's dismay, naive users (not unintelligent — just not in on the joke) were convinced of the wisdom of the advisor. The user's ability to make sense of signals where none truly resides may appear extreme, but it is by no means unusual. Collins, Green and Draper[7] tell of an automated counselor that provided totally random yes/no replies to questions posed by students; users were often able to make sense of the exchange.

It is undesirable to produce a display of data that contains no meaning and to rely entirely on the user to make sense of the scene by supplying the context and preconceptions necessary to interpretation. Graphics displays, of course, are not subject to this objection. They are closer to the opposite extreme, where meaning is contained in the display but where substantial expertise on the user's part is needed for a successful interpretation of the message. Instructions for assembling toys and preparing a meal from recipes are familiar examples of this situation. At the limit in which only minimal (though seldom zero) expertise is required of the user lie algorithms, almost all computer programs, and formal symbolic systems generally, considering the computer as the interpreter of the message.

In molecular modeling with computer graphics, users' expertise will vary widely, and the expert system's job will be to bridge the gap between the image displayed on the screen and the ideas that are meaningful to the user. The expert system should act as a guide rather like the docent at an art gallery, showing us how to look at an image. We can assume that the user sees the actual lines, colors and dots. What is perhaps most important is for the user to be helped to see 3D "shapes" accessible to the molecule. The problem of deducing shapes in three dimensions from a 2D image is precisely the problem in the subdiscipline of AI called computer vision.[8] Owing to its importance in surveillance, robotics, and manufacture, this application is under intense scrutiny.[9]

CHOOSING A TASK

Buchanan *et al.*[5] provide a number of maxims for constructing an expert system. Before we invest effort in developing an expert system, we must choose a reasonable task for which an expert system has a real chance of competence. In particular, we should choose a narrow specialized task that does not require a lot of commonsense knowledge. The density of context that makes interpreting written or oral natural language or images so easy for people is very difficult to make available to a computer program.

When choosing a task, you should:

- Make sure the task is neither too easy nor too hard for human experts, since we will wish to test the expert system's performance. "Too easy" means that it requires a few hours' work; machine assistance will hardly seem necessary. "Too hard" means a few months' work is needed; in this case it will be inconvenient to generate enough case studies, which are needed to deduce rules and to train the system.
- Define the task clearly. This is perfectly self-evident, but surprisingly easy to overlook in almost any problem.
- Use an expert. The practitioner is the keystone, providing the heuristics, test cases and explanations. However, the expert cannot be appreciated unless the designer of the expert system has also become familiar with the problem. The designer must be willing to become the apprentice.
- Record problem-solving sessions in which the expert explains reasoning and reflects on approaches, choices and hunches. From the transcript, you can identify and characterize key aspects of the problem, especially those that permitted the expert to make a decision.

Traditional expert systems have a knowledge base and an inference engine. A number of expert systems "shells" are available; in general, they use the inference engine from MYCIN[10] and require only that the designer supply rules. To establish the knowledge base, you must represent the information drawn from interviews with the expert. This may be difficult if the expert does not express decisions in MYCIN's "production-rule" form. Not all domains of knowledge lend themselves to MYCIN-style rules. These rules of thumb do not usually incorporate any "deep knowledge" — recognition of the constraints of physics and chemistry — which might provide some insurance against perfect nonsense and which might supply better explanations of expert system conclusions.

A CARICATURE OF EXPERT SYSTEM DESIGN

It is often informative to develop a very simple expert system, incorporating some semblance of the eventual problem. For purposes of discussion let's look at a toy problem,[11] jigsaw puzzle assembly. To my knowledge, this has never been directly addressed in AI research, though assembly occurs in many contexts. I choose jigsaw assembly because there are some extremely helpful analogies between this familiar process and molecular assembly. For example, docking bears some similarity to the process of adding a piece to an existent fragment of a puzzle.

Jigsaw puzzle assembly is a 2D problem, which removes some of the most difficult features of molecular assembly. This simplification can be removed by considering the type of puzzle in which a cube or sphere is assembled from cut blocks. More closely analogous problems are provided by folding Rubik's snake or origami.

Figure 1. A sample puzzle that challenges the JIGSAW assembler. Each PIECE has two, three or four SIDEs with a standard TAB or BITE. The subproblem of assembling PIECES of type EDGE is done first, and assembly is guided by matching SIDE form and the LINE character or PROPERTY. Adapted from Matisse Line Drawings and Prints, *Dover Publications, New York, 1979 ("Loulou in a flowered hat," etching, 1914)*

Shape-matching is the key feature of puzzle-solving, but very often the shape-matching occurs at a late stage, after color- and pattern-matching has occurred. The color match is a kind of long-range sorting so that a small number of promising candidates can be selected before the time-consuming shape-matching takes place. The hierarchy of molecular-matching that leads eventually to docking a fragment is a natural part of rapid and expert assembly of jigsaw puzzles.

To construct a puzzle-assembler expert system, we need a data representation and production rules for the assembly of a simplified jigsaw puzzle. We will use the data structure suggested by the simplest form of LISP, the list. We assume that each piece can be considered a (roughly square) quadrilateral with at least two sides modified by a tab (OUT) or a bite (IN). The entire puzzle is rectangular with L pieces making up each of the long sides and W pieces making up each of the short sides. There are four corner pieces, and $2(W\text{-}2) + 2(L\text{-}2)$ other (edge) pieces making up the frame. It is natural to represent the eventually assembled puzzle by an L x W array of PIECES. Figure 1 shows a sample puzzle that challenges the JIGSAW assembly.

Every PIECE is represented by a list of SIDES, in clockwise order starting from the bottom. Thus, STRAIGHT.OUT.IN.OUT would represent an edge piece that is resting on its straight side, and STRAIGHT.STRAIGHT.IN.OUT a corner piece

oriented as if it were at the lower left of the frame, or OUT.OUT.IN.OUT an interior piece. Reorienting a single piece corresponds to a cyclic permutation of sides.

Each side may be attributed a number of PROPERTIES, which will include LINE (present or absent) and perhaps later such refinements as various COLOR or PATTERN values present on that boundary. A plausible heuristic, which we will express as a production rule, is that lines (and colors and patterns) flow from one piece to its neighbor across sides that fit, rather than terminate at a side.

Now that we can represent a single piece, we must represent the fragments made up of several PIECEs. Almost all jigsaw puzzles are bounded, which is an important simplification — all puzzle solvers use the edge pieces as a constraint and subproblem, which is particularly important for small puzzles. First, let's consider the construction of the frame. We need only deal with EDGE pieces, which we can assemble into four linear clusters before forming the rectangular frame (i.e., the top and bottom rows and the rightmost and leftmost columns of the array representing the assembled puzzle. We begin by selecting a pair of single EDGE pieces positioned with the STRAIGHT edge downward, to test for a join. We need not generate-and-test all such pairs; we could instead prepare a collection of candidates for pairing that share a LINE (or COLOR, or PATTERN) status across sides. Some such heuristic restriction of the search space is a common feature of expert systems.

If we find a match, for example, between STRAIGHT.OUT.OUT.IN and STRAIGHT.OUT.IN.OUT, the two-piece cluster is represented by the array [STRAIGHT.OUT.OUT.IN-FIT][STRAIGHT.OUT-FIT.IN.OUT]. FIT is a property of a SIDE; it can be a pointer to a side of the next PIECE in the fragment array. Once the FIT pointer is assigned to a SIDE, we need not inspect that side again. But we can now test every isolated piece against the cluster's free sides and permit the cluster to grow. When no more of the isolated pieces in the collection of those held to be promising because of shared edge properties can add to a particular cluster, we begin another cluster.

We expect that assembly within sorted sets of edge pieces will be efficient but not perfect. In any event, there will be a number of clusters of substantial size, along with perhaps single pieces for which our side-property sorting gives little help, which now must be joined. The "gestalt" integration in which substantial fragments are joined and produce great leaps in the assembly is an especially satisfying feature of the process for the assembler. A weak echo of this might be expressed in a measure of progress — the mean-square of pieces per fragment, perhaps. (We will know when the task is finished because this measure becomes the square of the total number of pieces.)

The edge-assembly procedure will produce four fragment arrays. According to interviews, veteran human assemblers incorporate the corner pieces in this stage and also orient the four edges (and some free pieces and fragments) according to pattern. This is very common when the pieces or fragments can be interpreted from the overall picture — hard puzzles at least delay this more global integration.

This toy system, intended to illustrate some of the features of design of an expert system capable of some degree of spatial reasoning, incorporates in principle the following aspects of a more serious expert system:

- The task is largely free of commonsense and human context.
- The search space is large in principle but can be strikingly reduced by a combination of rigorous and heuristic constraints.
- Heuristics can be expressed as "production rules."
- There is a measure of progress toward the completion of the task.
- The task is eased by the need to make sense of the scene on the large scale.
- Unusual (but by no means unprecedented) among expert systems, there is a degree of geometric modeling.

SPECIAL REQUIREMENTS OF AN EXPERT SYSTEM IN MOLECULAR MODELING

Images of objects in 2D space carry properties quite different from the verbal-logical information processed in most currently useful expert systems. According to Riseman and Hanson[12] an expert system capable of interpreting scenes must

- incorporate a representation of "image events"; in our toy, these are PIECES
- possess low-level processes that extract image features; in our toy, these are the attributed PROPERTIES of COLOR, LINE, PATTERN
- employ sufficiently flexible and powerful 3D representations to capture geometrically complex objects
- have ways to use knowledge during interpretation; these have been left out of our toy

IMAGE EVENTS AND FEATURES

By "image events" the computer vision specialist means a small number of features of the display, such as lines, contours, regions of homogeneous color or texture, simple shapes, and (parts of) surfaces. While a typical 2D view of a scene might have an array of 512 x 512 pixels, the number of features would be much smaller. These features need further interpretation. The novice user of a molecular modeling system needs not just the lines (which are immediately apparent) but chemically and geometrically more meaningful clusters. It is entirely the reinterpretation by and for the viewer that is our present concern. Central to that reinterpretation, and the process which preoccupies most AI-vision research, is the inference of depth.

The 2D image is underconstrained, and the problem of the viewer — to construct at least some of the 3D structure of the object from the image — seems formidable indeed. Of course, human vision solves similar problems rapidly and effortlessly in everyday life, with

only rare intrusion into consciousness of ambiguity. The human visual system does fail, however, when it cannot bring expectations and context to bear on the problem. Galileo's drawing of Saturn's rings as essentially 2D "horns" shows that the most gifted observer can fall into the trap of unfamiliarity. The Necker cube illusion (and many more) works mainly because there is so little information on depth in a 2D projected line drawing. The viewer, insisting on some 3D interpretation, encounters a disturbing ambiguity. The plane surfaces are imputed one of (usually) two equally convincing orientations; you can consciously choose either but not both. A molecular representation by a 2D projection of the bond vectors is prone to such ambiguity, since — to the novice — there is little context.

Investigators of computer vision and graphics have identified a number of clues to depth. Some of these are easy to incorporate into a line-drawing display. Occlusion is perhaps the most important clue to depth. Removing occluded portions from the line drawing (leaving explicit occluding contours) is effective in resolving much depth ambiguity. Foreshortening and perspective is a time-tested means of suggesting the third dimension in a single 2D image.

Intensity (brightness) may be an important depth cue in real-life vision; distant objects appear more dimly lit than nearby objects. This is not highly sensitive — dimness varies slowly with distance — but can cue large differences, such as what edge of a molecular ring is directed toward the viewer; the more brightly drawn lines lie closer. Surface contours permit us to judge surface orientations, which suggest local depth differences and shapes as well. Clipping (erasing) portions of the line drawing that are farther from the viewer than some limit provides a crude indication of depth.

A line drawing rotated about an axis permits a more detailed and reliable "object hypothesis" than any static line drawing. You can achieve an illusion of depth with some economy if you choose a few distinct planes for the image. In "Snow White," distant trees, nearby furniture, and figures in the foreground are drawn on different transparencies; the more distant objects move more slowly as the point of view changes. Stereopsis is the classic explanation for depth perception; our two eyes provide slightly different views of an object. But stereopsis provides quite local cues to relative depth. In fact, most of these cues establish depth on a very local level. We cannot judge the relative depth of regions in quite different regions of the visual field.

Our human visual system may get important depth or shape information from all these features; images lacking these features may be hard to interpret. Theories of vision that emphasize these cues to depth may be called "bottom-up." But other theories of our attribution of a third dimension to an image emphasize our experience in a 3D world.[13] Our intimate and tactile acquaintance with a large vocabulary of arrangements in three dimensions may mean that the 2D image need only remind us of some familiar object or situation in order that we perceive — or recall — the third dimension.

The highly influential effort by Brooks[14] in the interpretation of 2D images by model-matching is particularly helpful here. Fortunately, the entire ACRONYM system devised by Brooks is not required, since the chemical model — of which only part is being displayed — has a known 3D structure. The system need not deduce any depth information. The viewer may be assisted to perceive the scene, including depth, from the top down if the expert system draws on the full depth information to help the user make a reasonable "object hypothesis." The expert system may suggest an object hypothesis for segments of the image, if it has at hand models that it can recognize in the coordinate data.

USE OF AN ACRONYM-STYLE MODELER

We should distinguish the chemical model — composed of atomic coordinates and conformational energy functions, electrostatic potentials and surfaces defined in three dimensions — from the models used by Brooks' ACRONYM, which are generalized cones. Generalized cones are primitive volume elements defined by a planar cross section, a space-curve "spine" and a sweeping rule. Vision researchers have noted the fact that complex shapes — skeletons of animals — can be quickly recognized from the few lines provided by pipe-cleaner models; the attenuation of the generalized cones is a useful caricature that simplifies the task of recognition. This type of simplification is familiar to chemists, since ball-and-stick molecular figures are constructed from generalized cones. The helix, ribbons, and CPK space-filling models are other simple examples.

Brooks' major contribution to our problem is the capacity to represent generic classes of objects. A range of generalized cylinders, from which a class of objects can be constructed, is represented by the range of parameters associated with the cross section and the sweeping rule in Brooks' program. Chemists will find this generality congenial, since we are accustomed to a language of forms, including such terms as alpha-helix, coil, and beta-turn, which embrace a class of structures rather than a highly specific structure.[15] These are indeed generalized cones (see Figure 2), though they're outside the set of forms Brooks defines.

Brooks achieves a representation of classes of objects by an unusual data structure, which we can adapt to a LISP-coded specification of a macromolecule with fair ease. A macromolecule can be described by a TREE, whose ROOT is the coarsest characterization of the molecule; for a branched chain, it would be the longest backbone. Even the longest backbone might be made up of segments linked by connections, so the root data structure would itself be a LIST. Sidechains would be specified by nodes lower in the TREE. The arc connecting the node to the ROOT, itself a LIST, would tell which segment of the ROOT is connected to the sidechain. Every generalized cylinder carries its local coordinates; the arc also tells the orientation of the sidechain to the segment. In each case (in our realization of the data structure), this auxilary information is stored as a PROPERTY.

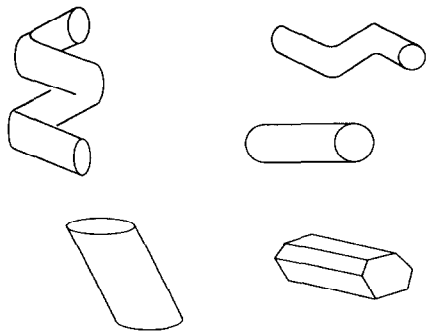

Figure 2. Sample generalized cones, defined by a cross section and a sweep rule. Forms include the COIL (circular cross section, restricted random walk sweeping rule), the HELIX (elliptic cross section, helical sweeping rule) and the RIGHT_CYLINDER (circular cross section, linear sweeping rule). Brooks also used polygonal cross sections and oblique linear sweeping rules

Figure 3. A ribbon diagram showing how complex figures can be enfolded by generalized cylinders (After M. Escher's "Bonds of Eternal Union")

We wanted our expert system to present to the viewer not the details of atomic coordinates or even the bond-vector projections now commonplace, but instead (or in addition), a model drawing wherein a simplified caricature of the structure, made of generalized-cone volume elements, represents the global structure. The display of such high-level representations of molecular structure is now quite advanced,[16] and cartoon displays are supplied by a number of vendors. These simplifications of bond-vector projections can reveal the global structure of a molecular forest without distracting the viewer with explicit strokes for every tree.

ASSISTING THE NOVICE

An expert guide can be used to select plausible first conformations of a macromolecule by interpreting the kinds of secondary structures available to a particular sequence of residues.[17,18] Alternatively, the expert system can be viewed as a means of delivery of expertise needed to interpret the model embodied in the molecular mechanics and atomic coordinates for the user. Our expert system, DOCENT, will present the molecule in caricature; viewing the caricature, it will be easier for the user to manipulate the molecule (i.e., select axes of rotation, recognize easy modes of vibration or segment rotation and perhaps recognize features of the molecular topography that typify hospitable docking sites).

One major problem facing users of the computer-generated graphic representations of macromolecules is the folding of the macromolecule into a form that permits an approaching fragment to optimize fit, into a cleft or active site. Here the problem of 3D assembly returns. The usual search for a receptor site ordinarily involves a statistical matching of a probe molecule's shape onto the accessible surface of a molecule of known geometry. Here we suggest an alternative approach. Using the caricature, which represents the molecule as a small collection of generalized cones, the user will find it easy to twist the model into a plausible form likely to provide a welcome to the probe. Molecular mechanics can give a hint to the cost of the folding.

Alternatively, the user can begin with the assumed shape of a substrate molecule and assemble a collection of generalized cones that accommodate the species. This may be considered a kind of automated assembly, to which Brooks has already applied his ACRONYM system.[19] There is no energetic criterion here, apart from the strictly geometric fit. But distance geometry is still useful, since it does not require that we deal with atoms, but can be extended to deal with shapes more generally,[20] by considering surface points along the generalized cylinders.

Once the collection of generalized cones is arranged in space to accommodate a prescribed shape, it may well be possible to find a sequence of residues that can give rise to the substructures implied by the cones.

A natural beginning for this program of work is the design of cyclic ethers that provide a receptive site for a variety of small organic materials.[21] Work on these systems was guided by physical CPK modeling, which computer graphics systems already ease.

CONCLUSION

Questioned about what unresolved difficulties are of the greatest urgency in AI research, Aaron Sloman[22] replied, "[A] major unsolved problem is how to represent shape. I believe that when we know how to represent shapes, spatial structures and spatial relationships, many other areas of AI will benefit, since spatial analogies and spatial modes of reasoning are so pervasive."

No discipline relies more thoroughly on the power of visualization in three dimensions than chemistry. Complex objects, which resist verbal and mathematical description stubbornly, can nonetheless be grasped visually. Chemists spend years developing the spatial intuition that informs their reasoning and speculation.

Expert systems incorporating sophisticated computer graphics can ease the development of this power and magnify it. We intend that DOCENT contribute to this end.

REFERENCES

1 Kollman, P. "Molecular modeling". *Annual Reviews of Physical Chemistry*, H. L. Strauss, Ed. Ann Rev Inc., Palo Alto, California, 1987, 301

2 Kolbe, H. *Journal fur Praktische Chemie*, Ser. 2, 1877, **15**, 473; see Snelders, H. A. M., Practical and theoretical objections to J. H. van't Hoff's 1874-stereochemical ideas, in *van't Hoff — LeBel Centennial*, O. B. Ramsay, Ed. American Chemical Society, Washington, DC, 1975, 55

3 Foley, J. D., and van Dam, A. *Fundamentals of Interactive Computer Graphics*. Addison-Wesley, Reading, Massachusetts, 1982

4 Cohen, P. R., and Feigenbaum, E. A. *The Handbook of Artificial Intelligence*, Vol. III. HeurisTech Press, W. Kaufmann, Los Altos, California, 1982

5 Hayes-Roth, F., Waterman, D. A., and Lenat, D. B., Eds. *Building Expert Systems*. Addison-Wesley, Reading, Massachusetts, 1983

6 Weizenbaum, J., *Computer Power and Human Reason: From Judgment to Calculation*. Freeman, W. H., San Francisco, California, 1976

7 Collins, H. M., Green, R. H., and Draper, R. C. "Where's the expertise? expert systems as a medium of knowledge transfer," in *Expert Systems 85*, M. Merry, Ed. Cambridge University Press, Cambridge, England, 1985, 325

8 Marr, D. *Vision*. W. H. Freeman, San Francisco, California, 1982

9 *Artificial Intelligence*, special issue on computer vision, 1981, **17**; a more recent review is given in D. G. Lowe, "Three-dimensional object recognition from single two-dimensional images", *Artificial Intelligence*, 1987, **31**, 355

10 Buchanan, B. G., and Shortliffe, E. H., Eds. *Rule Based Expert Systems*. Addison-Wesley, Reading, Massachusetts, 1984

11 Discussion based on work by Roman Grochowski, summer 1987, report for INST 394, University of Virginia; the suggestion was drawn from the description of an interpreter of line drawings by D. L. Waltz, "Understanding line drawings of scenes with shadows" in *Psychology of Computer Vision*, P. H. Winston, Ed., McGraw-Hill, 1975; summarized and interpreted in M. Boden, *Artificial Intelligence and Natural Man*, Basic Books, 1977, 225ff

12 Riseman, E. M., and Hanson, A. R. "A methodology for the development of general knowledge-based vision systems," in *Vision, Brain and Cooperative Computation*, M. A. Arbib and A. R. Hanson, Eds., MIT Press, Cambridge, Massachusetts 1987

13 Stevens, K. A. "Representing and analyzing surface orientation," in *Artificial Intelligence: An MIT Perspective*, Vol. 2, P. H. Winston and R. H. Brown, Eds. *MIT Press*, 1979, 101; "Recognition of shape in visible surfaces" by H. K. Nishihara in *Physical and Biological Processing of Images*, O. J. Braddick and A. C. Sleigh, Eds. Springer-Verlag, 1983, 335

14 Brooks, R. A. "Symbolic reasoning among 3-D models and 2-D images", *Artificial Intelligence*, 1981, **17**, 285

15 Richardson, J. S. "The anatomy and taxonomy of protein structure", *Advances in Protein Chem.*, 1981, **34**, 167

16 Burridge, J. M., and Todd, S. J. P. "Protein secondary structural representations using real-time interactive computer graphics", *J. Mol. Graph.*, 1986, **4**, 220

17 Leach, A. R., Dolata, D. P., and Prout, C. K. "AI in conformational analysis." Molecular graphics abstracts., *J. Mol. Graph.*, 1987, **2**, 107

18 Rawlings, C. J., *et al.* "Reasoning about protein topology using the logic programming language PROLOG." *J. Mol. Graph.*, 1985, **4**, 151

19 Brooks, R. A. "Model-based interpretations of three-dimensional interpretations of two-dimensional images." International Joint Conferences on Artificial Intelligence, 1981, **7**, 619

20 Crippen, G. M. *Distance Geometry and Conformational Calculations*. Research Studies Press of Wiley, New York, 1981

21 Cram, D. J., and Cram, J. H. "Design of complexes between synthetic hosts and organic guests", *Accounts of Chemical Research*, 1978, **11**, 8

22 Sloman, A., in the panel discussion "Artificial intelligence — where we are?" D. G. Bobrow and P. J. Hayes, Eds. *Artificial Intelligence*, 1985, **25**, 386