REPRESENTATION AND CONTROL IN VISION(*)

by

Aaron Sloman, David Owen, Geoffrey Hinton, Frank Birch, Frank O'Gorman
Cognitive Studies Programme,
School of Social Sciences,
University of Sussex,
Brighton, U.K.

(*) The Project is funded by the U.K. Science Research Council on grant no B-RG-8688-7. Frank O'Gorman has now left the group.

Is AI vision research making good progress?

Vision work in AI has made progress with relatively small problems. We are not aware of any system in which many different kinds of knowledge co-operate. Often there is essentially one kind of structure, e.g. a network of lines or regions, and the problem is simply to segment it, and/or to label parts of it. Sometimes models of known objects are used to guide the analysis and interpretation of an image, as in the work of Roberts (1965), but usually there are few such models, and there isn't a very deep hierarchy of objects composed of objects composed of objects ... By contrast, recent speech understanding systems, like HEARSAY (Lesser 1977, Hayes-Roth 1977), deal with more complex kinds of interactions between different sorts of knowledge. They are still not very impressive compared with people, but there are some solid achievements. Is the lack of similar success in vision due to inherently more difficult problems?

Some vision work has explored interactions between different kinds of knowledge, including the Essex coding-sheet project (Brady, Bornat 1976) based on the assumption that provision for multiple co-existing processes would make the tasks much easier. However, more concrete and specific ideas are required for sensible control of a complex system, and a great deal of domain-specific descriptive know-how has to be explicitly provided for many different sub-domains.

POPEYE and HEARSAY

The POPEYE project is an attempt to study ways of putting different kinds of visual knowledge together in one system. Our philosophy has much in common with HEARSAY, including:

(a) using strategies which may not find optimal solutions, but behave sensibly in a non-trivial set of tasks, in a "friendly" world,
(b) trying to find a good global interpretation without necessarily analysing and recognising all the relevant substructures,
(c) working outwards from the "best" (e.g. biggest, least ambiguous) fragments, at any level,
(d) commitment to distributed processing, since expertise often depends

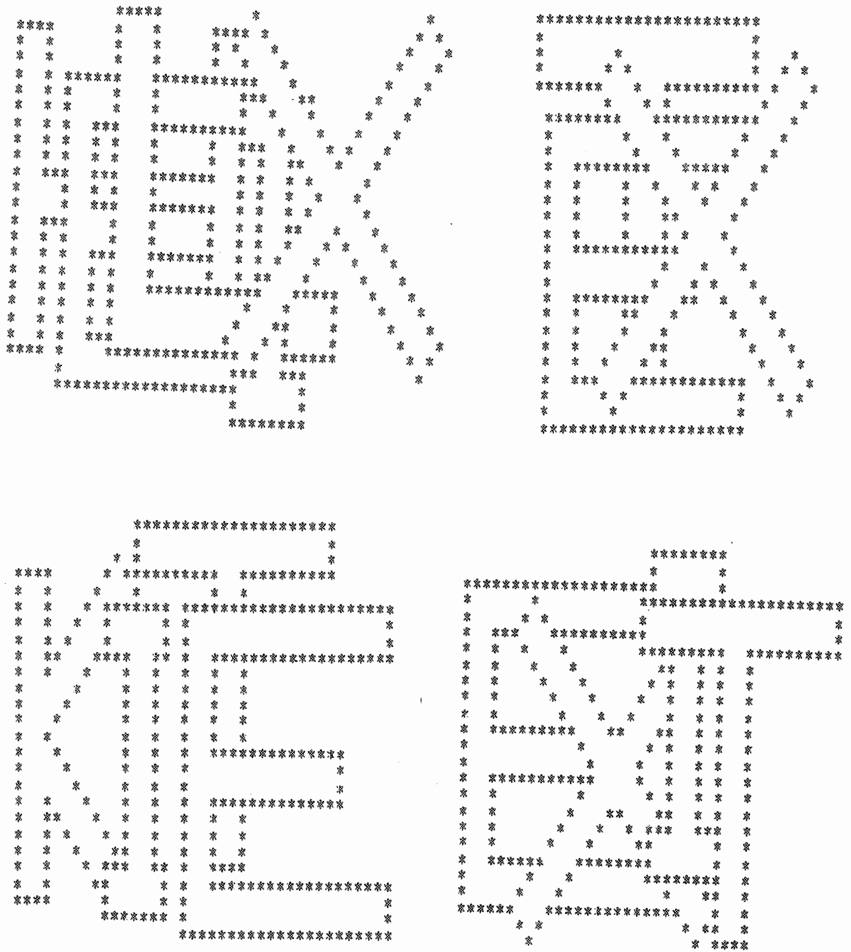
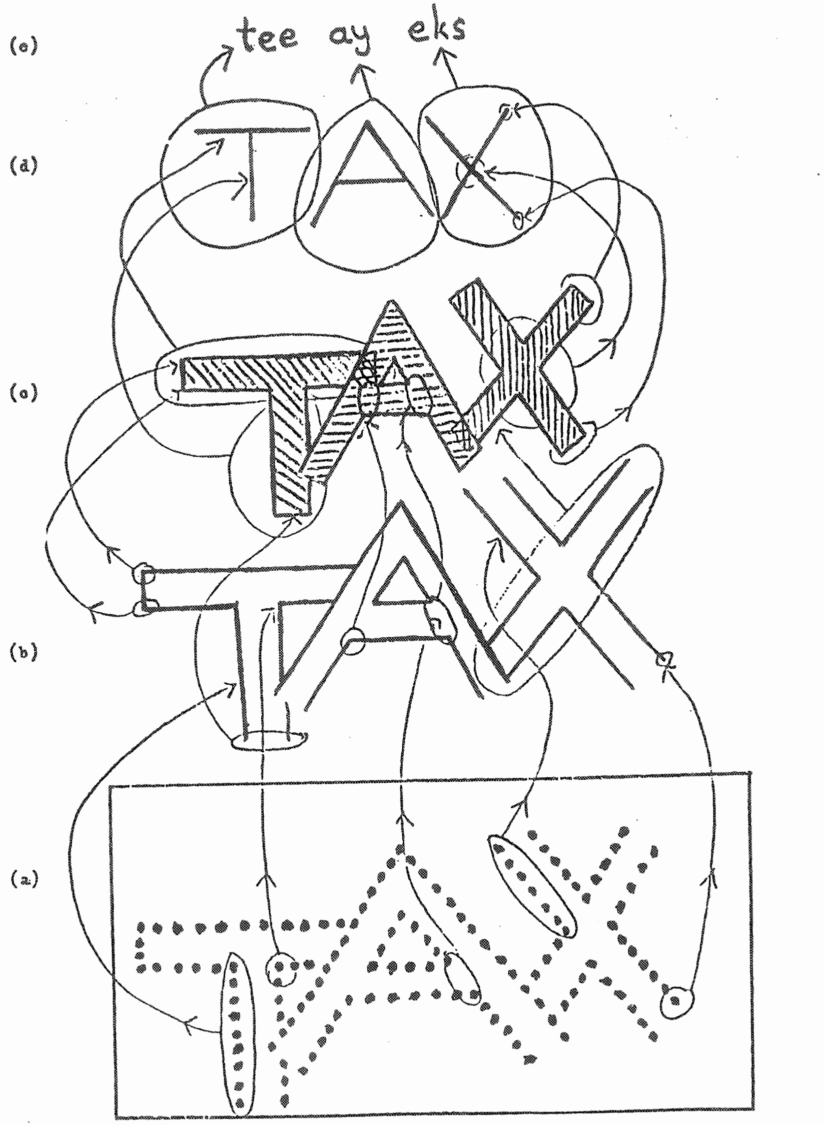


Figure 1

Examples of the pictures POPEYE analyses input as 2-D binary arrays. (Positive and negative noise can be added.) The more obvious lines, bars, bar-junctions and letters are found by the program. Current performance will be illustrated with slides at the conference.



Domains of interpretation of POPEYE pictures: - (a) configurations of dots, spaces, dot-strips etc., (b) configurations of line-segments, gaps, junctions, etc., (c) configurations of plates involving bars, bar-junctions, overlaps, edges of bars, etc., (d) stroke configurations, (e) letter sequences.

on conducting parallel searches in different domains, with partial results in one transforming the search-space in others,

(e) ordering different domains, and associated processes in a priority hierarchy, for scheduling scarce resources,

But POPEYE differs from HEARSAY. Where HEARSAY uses a general-purpose database (called a "black-board") with monitors to activate sub-systems when the contents are changed, POPEYE builds a variety of different kinds of networks (including 2-D arrays and other "analogical" representations), tailored to suit what they represent, so that, for example, searches are easily constrained by using the structure of the representation.

HEARSAY's modularity depends on uniform database procedures, whereas we achieve adequate modularity by making all sub-systems which access a particular type of sub-network use a procedure which knows how to deal with that kind of structure and knows which other sub-systems to invoke as a result. Much of the modularity of fixed-format data-base assertions can be achieved using fixed format procedure calls. Dynamic changes in such procedures are easily programmed by making them access special lists. Dynamic binding of function variables in POP2 also helps (Anderson 1976).

HEARSAY subsystems are supposed to know very little about one another, whereas, in POPEYE, messages (functions to be executed - Sloman & Hardy 1976) are sent explicitly from one sub-system to another (usually via a scheduler). We factor procedures according to whether they are concerned only with one kind of sub-system, or whether they are concerned with a message from one to another. Contrast the "frames" approach, i.e.: attach each chunk of procedural knowledge to one class of frames.

HEARSAY schedules tasks according to a numerical evaluation function combining many different measures, whereas our scheduling is based on a simple qualitative partition of sub-tasks into priority ordered classes, each with its own scheduling strategy, defaulting to a simple queue. E.g. when trying to find a word in pictures like figure 1, order tasks according to whether they are concerned with a word, a letter, a junction between bars, a bar, a line-segment, scanning for dots, etc. Choose the longest first, among line-segment jobs. Use a different ordering for a different goal, like find "all the vertical bars". So the mapping between priority levels and conceptual levels is represented by the order of a list of job-categories, and is therefore very easily altered. It is not obvious how task-dependent re-ordering of the levels in HEARSAY is achieved. Similarly a new category of sub-tasks is very easily spliced into POPEYE's priority list. The scheduling criteria used within different sub-categories need not be comparable. Long lines are (usually) more important than short ones, but the system never has to weigh length of lines against size of letters. This is consistent with what might happen in a parallel implementation.

Thus, the main scheduler selects the best jobtype, then its manager selects the best job. This makes it relatively easy to produce some major qualitative changes in performance without having to assess the interactions of several different weighting calculations. Other changes are harder to achieve without a numerical evaluation system, e.g. making the system give high priority to finding very short horizontal lines near isolated dots and large diagonal bars. But it is not clear that people can easily cope with arbitrary and complex attention-focussing instructions either.

Top-down processing in HEARSAY uses goals added to the data-base which trigger relevant procedures. In POPEYE (as in people?) much top-down processing involves a higher-level process directly sending a request to a lower-level to do its stuff in a particular region of the image space, with some of its criteria modified (e.g. if a horizontal line is wanted, then the dot-scanner should be less fussy about accepting evidence for horizontal lines in the specified region.) Again, HEARSAY allows more sensitive scheduling, whereas POPEYE is simpler and cheaper to do. The simplicity makes it more likely that a program will one day be able to do task-dependent re-scheduling. It is not obvious what the trade-offs are. Much depends on the kind of world the program is expected to cope with, on the tasks it is to perform in that world, and the costs of various kinds of errors and delays.

HEARSAY uses searching mechanisms able to cope with co-existing competing hypotheses, whereas our philosophy is to assume that when one cannot choose between alternatives one should describe what is common to them (cf. discrimination nets), or ignore them, then work on some other sub-task, hoping that emerging context will resolve the ambiguity. E.g. if relations between ambiguous limb-like shapes may unambiguously identify a human figure, then don't waste resources generating both leg and arm hypotheses for all the shapes (Paul 1976). Simpler searching is traded against more elaborate structural descriptions, in more domains (figure 2). This approach cannot cope with really difficult situations e.g. puzzle pictures. Where more problem-solving power is needed, and not merely domain-specific expertise, we are experimenting with the mechanism described by Birch and a relaxation mechanism for combinatorial search controlled by "preferences". (Hinton 1976, 1977).

Domain-specific representations

Because of the huge amounts of data in visual images and their interpretations, specialised representations are needed, so that access to information is rapid, and mutually relevant fragments may be linked to form significant cues without enormous combinatorial searches.

E.g. on finding a dot in the array, POPEYE can quickly check whether the local context is messy and ambiguous, since neighbouring dots are readily accessible. Similarly, the picture is mapped into an array of "zones" each of which knows about important image or scene fragments located there. Every "orientation" stores known ("infinite") lines in that orientation, in an ordered chain, so that each can quickly find its neighbours, and POPEYE can quickly examine all lines between two given lines. Each line lists the important line-segments and gaps which lie along it, ordered as they occur in the picture. A line-segment knows about bar-walls hypothesised as lying on it. Two bars may share a segment, one on each side. Each bar wall knows about the bar it is a wall of, and about the segment depicting it. A bar knows about its walls, its major axis, the junctions with other bars with which it merges, etc. Many image and scene structures have an intrinsic direction-system associated with them, défining a front end and a back end, a left-side and a right-side, etc. This is used by orientation-independent programs for manipulating them.

Problems of noticing significant substructures whilst such a network grows are non-trivial, and so far we have only partial solutions. Birch

(1978) describes one approach, combining recognition and segmentation, in a self-improving discrimination net.

Different categories of sub-processes distinguished by the scheduler correspond roughly to a variety of "intermediate" representations, storing partial results of processing, minimising the need for backtracking, i.e. allowing structure-sharing between different searches. (Marr (1976) calls this 'The principle of least commitment'. Cf. Sloman and Hardy 1976 p 252.) We have to examine many image and scene-fragments to discover the significant kinds of structures and relations. The hardest task is finding good descriptions, especially descriptions capable of adequately representing incompletely analysed structures, the "intermediate" representations. This study of the structure of the task domain is analogous to linguistic studies of the grammar and semantics of languages. The need to cope with partial information in a performing system leads to a richer ontology than a study of "competence".

Choosing a domain for studying vision

We are studying a variety of different sorts of images and scenes. We started with pictures like those in figure 1 for various reasons, including:

(a) People can interpret them (and can get better with practice), so there is a relevant human ability to be explained.

(b) Progressively more difficult examples can be produced, with positive or negative noise and confusing occlusions and juxtapositions, to test the program's ability to degrade gracefully.

(c) As shown in figure 2, sensible interpretation of the pictures requires knowledge about structures in different domains.

(d) Some of the image structures also arise in pictures of 3-D blocks-world scenes.

(e) Information flowing across domains enables redundancy to be used to find a good global interpretation without processing all details.

(f) Computer-generated pictures enable us to do some useful work despite the lack of TV equipment and a shortage of space in our computer.

What is heterarchy?

POPEYE is a step towards a heterarchic system which could be implemented on distributed processors. Some take "heterarchy" to refer to the use of fancy control structures. We think it is a negatively defined concept, contrasted with hierarchy. A hierarchical perceptual system is one which can be represented as a linear pipeline of processes thus:

Input ---> P1 ---> P2 ---> P3 ---> P4 ---> Output

Where arrows represent flow of data and, in a serial processor, control. A simple-minded alternative is to permit feed-back loops between sub-processes, so that not all the arrows go one way. A more complex alternative is to postulate several parallel pipe-lines through which information can flow, with additional routes for possible feed-back, feed-across, and feed-forward, as in figure 3 (optional inputs represent prior expectations). Each Pij may have several sub-processes active at different locations in the picture - (See figure 2).

In POPEYE there is a general drift of information along each pipeline and from lower pipelines to higher ones, but with occasional short-circuits. Each layer is concerned with a different domain (figure 2). Within a layer, or pipeline, information tends, on the whole, to flow from smaller to larger (higher-priority) structures. But this is only a default, over-ridden by detection of significant cues or the use of prior expectations and higher-level knowledge, causing hypotheses or requests to be formulated which can flow up or down between layers, or skip backwards or forward within a layer.

So recognition of some fragment can make the system jump to a more global hypothesis about a larger whole containing it, or an interpretation in a different domain. A high-level cue may invoke the correct global interpretation while much lower-level processing is still incomplete. The effects of such knowledge-based jumps in processing are, we believe, more important than the advantages gained from using general-purpose mathematically complete or even optimal search strategies within any one problem-space.

Short Bibliography

Anderson, Bruce 'A brief critique of LISP' in Proc. AISB Conf. 1976.

Birch, Frank et al, 'A (self adapting) network for recognition of visual structures' AISB 1978.

Bornat, R. 'Reasoning about hand printed FORTRAN programs' in AISB 1976.

Bornat, R. & J.M. Brady, 'Finding blobs of writing in the FORTRAN coding-sheets project' Proc. AISB Conf. 1976.

Brady, J.M. & B.J. Weilinga 'Seeing a pattern as a character', in Proc. AISB Conf. 1976.

Hayes-Roth F, & V.R. Lesser, 'Focus of Attention in the HEARSAY-II Speech Understanding System', in 5th IJCAI, 1977.

Hinton G. 'Using relaxation to find a puppet', in Proc. AISB Conf. 1976.

Hinton G. PhD thesis, Edinburgh University AI Dept. 1977.

Lesser V.R. & L.D. Erman, 'A retrospective view of the HEARSAY-II Architecture', 5th IJCAI 1977.

Marr, D. 'Early processing of visual information', in Philosophical transactions of the Royal Society of London, pp 483-519 1976.

Paul J.L. 'Seeing Puppets quickly', Proc. AISB Conf. 1976.

Sloman A. & S. Hardy 'Giving a computer Gestalt Experiences', AISB 1976.

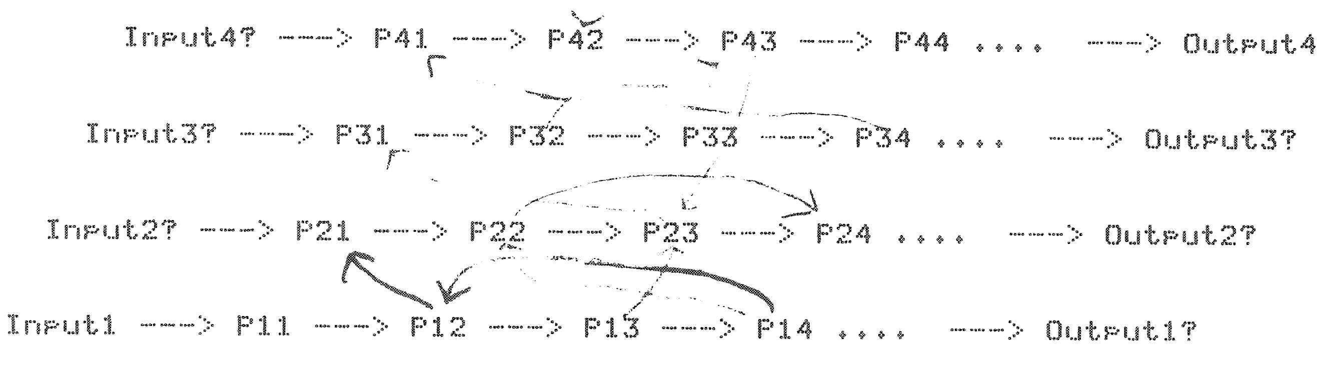

Input4? ---> P41 ---> P42 ---> P43 ---> P44 .... ---> Output4
Input3? ---> P31 ---> P32 ---> P33 ---> P34 .... ---> Output3?
Input2? ---> P21 ---> P22 ---> P23 ---> P24 .... ---> Output2?
Input1 ---> P11 ---> P12 ---> P13 ---> P14 .... ---> Output1?

Figure 3