NOTE ADDED 21 Jul 2015
Since July 2015 this file is out of date.
The completely repackaged book can now be found here in html and pdf versions:
http://www.cs.bham.ac.uk/research/projects/cogaff/crp/crp.html
http://www.cs.bham.ac.uk/research/projects/cogaff/crp/crp.pdf

The Computer Revolution In Philosophy (1978)

Aaron Sloman

Book contents page

This chapter is also available in PDF format here.

CHAPTER 9
PERCEPTION AS A COMPUTATIONAL PROCESS

9.1. Introduction

In this chapter I wish to elaborate on a theme which Immanuel Kant found obvious: there is no perception without prior knowledge and abilities.

In the opening paragraphs of the Introduction to *Critique of Pure Reason* he made claims about perception and empirical knowledge which are very close to assumptions currently made by people working on artificial intelligence projects concerned with vision and language understanding. He suggested that all our empirical knowledge is made up of both 'what we receive through impressions' and of what 'our own faculty of knowledge supplies from itself. That is, perception is not a passive process of receiving information through the senses, but an active process of analysis and interpretation, in which 'schemata' control what happens. In particular, the understanding has to 'work up the raw material' by *comparing* representations, *combining* and *separating* them. He also points out that we may not be in a position to distinguish what we have added to the raw material, 'until with long practice of attention we have become skilled in separating it'. These ideas have recently been re-invented and elaborated by some psychologists (for example, Bartlett).

One way of trying to become skilled in separating the raw material from what we have added is to attempt to design a machine which can see. In so doing we learn that a great deal of prior knowledge has to be programmed into the machine before it can see even such simple things as squares, triangles, or blocks on a table. In particular, as Kant foresaw, such a machine has to use its knowledge in comparing its sense-data, combining them into larger wholes, separating them, describing them, and interpreting them as representing some other reality. (This seems to contradict some of the claims made by Ryle about perception, in his 1949, e.g. p. 229, last paragraph.)

[[Note added August 2002:
A slide presentation on requirements for some sort of "innate" conceptual information in intelligent systems can be found here
http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk14
*Getting meaning off the ground: symbol grounding vs symbol attachment.***]]**

[[Note added Jan 2007
During 2005-6, while working on the CoSy robotic project I became increasingly aware that the ideas presented here and in several later papers were too much concerned with perception of multi-layered **structures**, ignoring perception of **processes**, especially concurrent perception of processes at different levels of abstraction. This topic was discussed in this presentation
"http://www.cs.bham.ac.uk/research/projects/cosy/papers/#pr0505
"A (Possibly) New Theory of Vision."
And in this much older paper:
Aaron Sloman, 1983, Image interpretation: The way ahead?, in
Physical and Biological Processing of Images
(Proceedings of an international symposium organised by The Rank Prize Funds, London, 1982.), Eds. O.J. Braddick and A.C. Sleigh, pp. 380--401, Springer-Verlag, Berlin,
http://www.cs.bham.ac.uk/research/projects/cogaff/06.html#0604 **]]**

9.2. Some computational problems of perception

People have very complex perceptual abilities, some of them shared with many animals. Especially difficult to explain is the ability to perceive form and meaning in a complex and messy collection of ambiguous and noisy data. For instance, when looking at a tree we can make out twigs, leaves, flowers, a bird's nest, spiders' webs and a squirrel. Similarly, we can (sometimes) understand what is said to us in conversations at noisy parties, we can read irregular handwriting, we can see familiar objects and faces depicted in cartoons and 'modern' paintings, and we can recognise a musical theme in many different arrangements and variations.

Seeing the significance in a collection of experimental results, grasping a character in a play or novel, and diagnosing an illness on the basis of a lot of ill-defined symptoms, all require this ability to make a 'Gestalt' emerge from a mass of information. A much simpler example is our ability to see something familiar in a picture like Figure 1. How does a 'Gestalt', a familiar word, emerge from all those dots?

Close analysis shows that this kind of ability is required even for ordinary visual perception and speech understanding, where we are totally unaware that we are interpreting untidy and ambiguous sense-data. In order to appreciate these unconscious achievements, try listening to very short extracts from tapes of human speech (about the length of a single word), or looking at manuscripts, landscapes, street scenes and domestic objects through a long narrow tube. Try looking at portions of Figure 1 through a hole about 3 mm in diameter in a sheet of paper laid on the figure and moved about. This helps to reveal how ambiguous and unclear the details are, even when you think they are clear and unambiguous. Boundaries are fuzzy, features indistinct, possible interpretations of parts of our sense-data indeterminate.

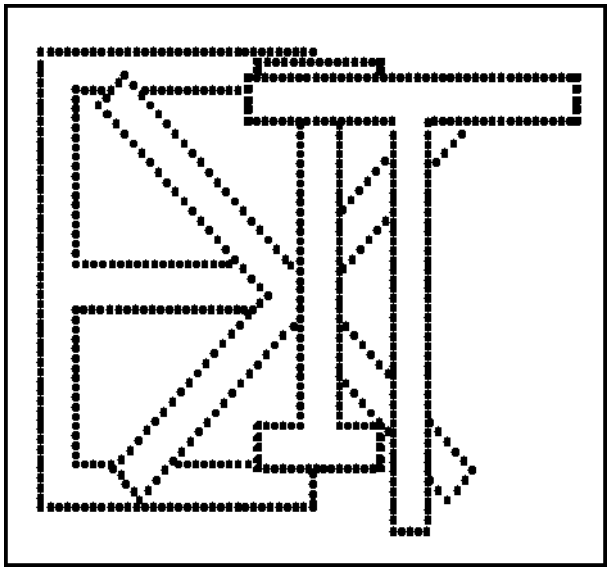


Figure 1

Fragments of this picture are quite ambiguous, yet somehow they help to disambiguate one another, so that most people see a pile of letters forming a familiar word. Often the word is seen before all the letters are recognized, especially if noise is introduced making recognition of the letters harder (e.g. if some dots are removed and spurious dots added). Without knowledge of letters we would have no strong reason to group some of the fragments, e.g. the top of the "I" and the rest of the "I".

Perceived fragments require a context for their interpretation. The trouble is that the context usually consists of other equally ambiguous, incomplete, or possibly even spurious fragments.

Sometimes our expectations provide an additional context, but this is not essential, since we can perceive and interpret totally unexpected things, like a picture seen on turning a page in a newspaper, or a sentence overheard on a bus.

9.3. The importance of prior knowledge in perception

What we can easily perceive and understand depends on what we know. One who does not know English well will not be able to hear the English sentences uttered at a noisy party, or to read my handwriting! Only someone who knows a great deal about Chemistry will see the significance in a collection of data from chemical experiments. Only someone with a lot of knowledge about lines, flat sheets, letters and words will quickly see ’EXIT’ in Figure 1.

Perception uses knowledge and expertise in different ways, clearly brought out by work on computer programs which interpret pictures. One of the most striking features of all this work is that it shows that very complex computational processes are required for what appeared previously to be very simple perceptual abilities, like seeing a block, or even seeing a straight line as a line. These processes make use of many different sorts of background knowledge, for instance in the following conscious and unconscious achievements:

a) Discerning features in the sensory array (for instance discerning points of high contrast in the visual field),

b) Deciding which features to group into significant larger units (e.g. which dots to group into line segments in Figure 1),

c) Deciding which features to ignore because they are a result of noise or coincidences, or irrelevant to the present task,

d) Deciding to separate contiguous fragments which do not really belong together (e.g. adjacent dots which are parts of the boundaries of different letters),

e) Making inferences which go beyond what is immediately given (e.g. inferring that the edge of one bar continues behind another bar, in Figure 1),

f) Interpreting what is given, as a representation of something quite different (e.g. interpreting a flat image as representing a scene in which objects are at different depths: Figure 1 is a very simple example),

g) Noticing and using inconsistencies in an interpretation so as to re-direct attention or re-interpret what is given.

h) Recognising cues which suggest that a particular mode of analysis is appropriate, or which suggest that a particular type of structure is present in the image or the scene depicted e.g. detecting the *(style* of a picture this can enable an intelligent system to avoid a lot of wasteful searching for analyses and interpretations.

So, perceiving structure or meaning may include using knowledge to reject what is irrelevant (like background noise, or coincidental juxtapositions) and to construct or hallucinate what is not there at all. It is an active constructive process which uses knowledge of the ’grammar’ of sensory data, for instance knowledge of the possible structures of retinal images, knowledge about the kinds of things depicted or represented by such data, and knowledge about the processes by which objects generate sense-data. Kant’s ’schemata’ must incorporate all this.

We need not be aware that we possess or use such knowledge. As Kant noticed, it may be an ’art concealed in the depths of the human soul’ (p. 183, in Kemp Smith’s translation), much of it "compiled" into procedures and mechanisms appropriate to images formed by the kind of world we live in. But at present there are no better explanations of the possibility of perception than explanations in terms of intelligent processes using a vast store of prior information, much of which is "compiled" (by evolution or by individual learning) into procedures and mechanisms appropriate to images formed by the kind of world we live in.

For instance, theories according to which some perception is supposed to be ’direct’, not involving any prior knowledge, nor the use of concepts, seem to be quite vacuous. A theory which claims that perceptual achievements are not decomposable into sub-processes cannot be used as a basis for designing a working mind which can perceive any of the things we perceive. It lacks explanatory power, because it lacks generative power. If the processes cannot be decomposed, then there is no way of generating the huge variety of human perceptual abilities from a relatively economical subsystem. By contrast, computational theories postulating the use of prior knowledge of structures and procedures can begin to explain some of the fine structure (see chapters 2 and 3) of perceptual processes, for example, the perception of this as belonging to that, this as going behind that, this as similar to that, this as representing that, and so on.

Quibbles about whether the ordinary word ’knowledge’ is appropriate for talking about the mechanisms and the stored facts and procedures used in perception seem to be merely unproductive distractions. Even if the ordinary usage of the word ’knowledge’ does not cover such inaccessible information, extending the usage would be justified by the important insights gained thereby. Alternatively, instead of talking about ’knowledge’ we can talk about ’information’ and say that even the simplest forms of perception not only provide new information, in doing so they make use of various kinds of prior information.

In a more complete discussion it would be necessary to follow Kant and try to distinguish the role of knowledge gained from previous perceptual experiences and the role of knowledge and abilities which are required for any sort of perceptual experience to get started. The latter cannot be empirical in the same sense, though it may be the result of millions of years of evolutionary "learning from experience".

Since our exploration of perceptual systems is still in a very primitive state, it is probably still too early to make any such distinctions with confidence. It would also be useful to distinguish general knowledge about a class of theoretically possible objects, situations, processes, etc., from specific knowledge about commonly occurring subsets. As remarked in chapter 2, we can distinguish knowledge about the *form* of the world from knowledge about its *contents.* Not all geometrically possible shapes are to be found amongst animals, for example. A bat may in some sense be half way between a mouse and a bird: but not all of the intervening space is filled with actually existing sorts of animals. If the known sorts of objects cluster into relatively discrete classes, then this means that knowledge of these classes can be used to short-circuit some of the more general processes of analysis and interpretation which would be possible. In information-theoretic terms this amounts to an increase of redundancy -- and a reduction of information -- in sensory data. This is like saying that if you know a lot of relatively commonly occurring words and phrases, then you may be able to use this knowledge to cut down the search for ways of interpreting everything you hear in terms of the most general grammatical and semantic rules. (Compare Becker on the ’phrasal lexicon’.) This is one of several ways in which the environment can be cognitively ’friendly’ or ’unfriendly’. We evolved to cope with a relatively cognitively friendly environment.

In connection with pictures like Figure 1, this means that if you know about particular letter-shaped configurations of bars, then this knowledge may make it possible to find an interpretation of such a picture in terms of bars more rapidly than if only general bar-configuration knowledge were deployed. For instance, if you are dealing with our capital letters, then finding a vertical bar with a horizontal one growing to the left from its middle, is a very good reason for jumping to the conclusion that it is part of an ’H’, which means that you can expect another vertical bar at the left end of the horizontal.

Thus a rational creature, concerned with maximising efficiency of perceptual processing, might find it useful to store a very large number of really quite redundant concepts, corresponding to commonly occurring substructures (phrases) which are useful discriminators and predictors.

The question of how different sorts of knowledge can most fruitfully interact is a focus of much current research in artificial intelligence. The strategies which work in a ’cognitively friendly world’ where species of things cluster are highly fallible if unusual situations occur. Nevertheless the fallible, efficient procedures may be the most rational ones to adopt in a world where things change rapidly, and your enemies may not give you time to search for a *conclusive* demonstration that it is time to turn around and run. Thus much of the traditional philosophical discussion of rationality, in terms of what can be proved beyond doubt, is largely irrelevant to real life and the design of intelligent machines. But new problems of rationality emerge in their place, such as problems about trading space against time, efficiency against flexibility or generality, and so on. From the design standpoint, rationality is

largely a matter of choosing among trade-offs in conditions of uncertainty, not a matter of getting things 'right', or selecting the 'best'. (For more on trade-offs see the chapters on representations, and on numbers: Chapter 7 and Chapter 8)).

9.4. Interpretations

Knowledge is used both in analysing structures of images and in interpreting those structures as depicting something else. There may be several different layers of interpretation. For example in Figure 1, dot configurations represent configurations of lines. These in turn represent configurations of bars. These represent strokes composing letters. And sequences of letters can represent words (see fig. 6). Within each layer there may be alternative structures discernible, for instance alternative ways of grouping things, alternative relations which may be noticed. These will affect the alternative interpretations of that layer. By examining examples in some detail and trying to design mechanisms making the different experiences possible we can gain new insights into the complex and puzzling concept of 'seeing as', discussed at length in part II of Wittgenstein's *Philosophical Investigations.*

Contrary to what many people (including some philosophers) have assumed, there need not be any similarity between what represents and what it represents. Instead, the process of interpretation may use a variety of interpretation rules, of which the most obvious would be rules based on information about a process of projection which generates, say, a two-dimensional image from a three-dimensional scene. (For more on this see the chapter on analogical representations.)

The projection of a three dimensional scene onto a two dimensional image is just a special case of a more general notion of *evidence* which is generated in a systematic way by that which explains it. A two-dimensional projection of a three-dimensional object bears very little similarity to the object. (Cf. Goodman, *Languages of Art.)* The interpretation procedure may allow for the effects of the transformations and distortions in the projection (as a scientist measuring the temperature of some liquid may allow for the fact that the thermometer cools the liquid).

This is an old idea: what is new in the work of A.I. is the detailed analysis of such transformations and interpretation procedures, and the adoption of new standards for the acceptability of an explanation: namely it must suffice to generate a working system, that is, a program which can use knowledge of the transformations to interpret pictures or the images produced by a television camera connected to the computer.

What we are conscious of seeing is the result of many layers of such interpretation, mostly unconscious, yet many of them are essentially similar in character to intellectual processes of which we are sometimes conscious. All this will be obvious to anyone familiar with recent work in theoretical linguistics.

So the familiar philosophical argument that we do not see things as they are, because our sense-organs may affect the information we receive, is invalid. For however much our sense-organs affect incoming data, we may still be able to interpret the data in terms of how things really are. But this requires the use of knowledge and inference procedures, as people trying to make computers see have discovered. Where does the background knowledge come from? Presumably a basis is provided genetically by what our species has learnt from millions of years of evolution. The rest has to be constructed, from infancy onwards, by individuals, with and without help, and mostly unconsciously.

9.5. Can physiology explain perception?

To say that such processes are unconscious does not imply that they are physiological as people sometimes assume in discussions of such topics. Physical and physiological theories about processes in the brain cannot account for these perceptual and interpretative abilities, except possibly at the very lowest levels, like the ability to detect local colour contrasts in the visual field. Such tasks can be delegated to physical mechanisms because they are relatively determinate and context-independent, that is algorithmic (e.g. see Marr, 1976). In particular, such peripheral processes need not involve the construction and testing of rival hypotheses about how to group fragments of information and how to interpret features of an image. But, physical and physiological mechanisms cannot cope with the more elaborate context-dependent problem-solving processes required for perception. The concept of using stored knowledge to interpret information has no place in physics or physiology, even though a physical system may serve as the computer in which information is stored and perceptual programs executed.

Moreover, even colour contrasts can sometimes be hallucinated on the basis of context, as so-called 'illusory-contrasts' show. For an example see Figure 2.

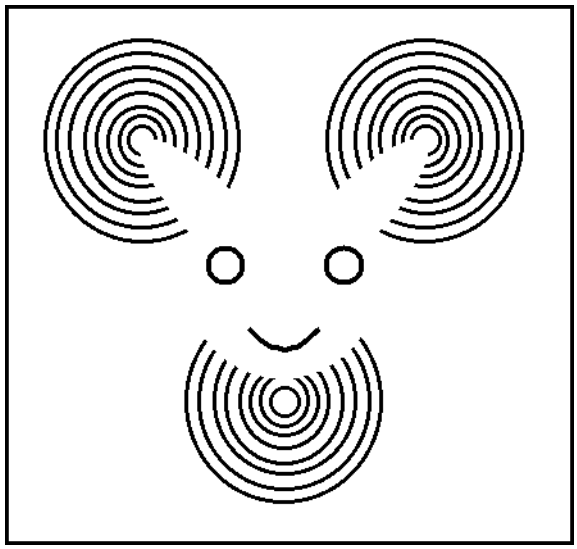


Figure 2

This picture (based on Kanizsa, 1974) shows that perceived colour at a location depends not only on the corresponding physical stimulus, hut also on the context. Most people see the central region as whiter than the rest, even though there is no physical difference.

Instead of physiological theories, we need 'computational theories, that is, theories about processes in which symbolic representations of data are constructed and manipulated. In such processes, facts about part of an image are interpreted by making inferences using context and background knowledge. We must not be blinded by philosophical or terminological prejudices which will not allow us to describe unconscious processes as inferences, or, more generally, as 'mental processes'.

How is it done? In particular, what exactly is the knowledge required for various kinds of perception, and how do we mobilise it as needed? We cannot yet claim to have complete or even nearly complete explanations. But A.I. work on vision has made some significant progress, both in showing up the inadequacies of bad theories and sketching possible fragments of correct explanations.

Our present ignorance is not a matter of our not knowing which theory is correct, but of our not even knowing how to formulate theories sufficiently rich in explanatory power to be worth testing experimentally.

Attempting to program computers to simulate human achievements provides a powerful technique for finding inadequacies in our theories thereby stimulating the rapid development of new theory-building tools. In the process we are forced to re-examine some old philosophical and psychological problems. For a survey of some of this work, see the chapters on computer vision in Boden (1977). Winston (1975) also includes useful material, especially the sections by Winston, Waltz, and Minsky. The rest of this chapter illustrates some of the problems with reference to an ongoing computer project at Sussex University, which may be taken as representative.

9.6. Can a computer do what we do?

We are exploring some of the problems of visual perception by attempting to give a computer the ability to perceive a configuration of known shapes in a scene depicted by a 'spotty' picture like Figure 1. The pictures are presented to the program in the form of a 2-dimensional binary (i.e. black and white) array. The array is generated by programs in the computer either on the basis of instructions, or with the aid of a graphical input terminal. Additional spurious dots ('positive noise') can be added to make the pictures more confusing. Similarly, spurious gaps ('negative noise') can be added.

People can cope quite well with these pictures even when there is a lot of positive and negative noise, and where further confusion is generated by overlaps between letters, and confusing juxtapositions. Some people have trouble at first, but after seeing one or two such pictures, they interpret new ones much more rapidly. The task of the program is to find familiar letters without wasting a lot of time investigating spurious interpretations of ambiguous fragments. It should 'home in on' the most plausible global interpretation fairly rapidly, just as people can.

Out of context, picture details are suggestive but highly ambiguous, as can be seen by looking at various parts of the picture through a small hole in a sheet of paper. Yet when we see them in context we apparently do not waste time exploring all the alternative interpretations. It is as if different ambiguous fragments somehow all 'communicate' with one another in parallel, to disambiguate one another.

Waltz (1975) showed how this sort of mutual disambiguation could be achieved by a program for interpreting line drawings representing a scene made up of blocks on a table, illuminated by a shadow-casting light. He gave his program prior knowledge of the possible interpretations of various sorts of picture junctions, all of which were ambiguous out of context. So the problem was to find a globally consistent interpretation of the whole picture. The program did surprisingly well on quite complex pictures. His method involved letting possible interpretations for a fragment be 'filtered out' when not consistent with any possible interpretations for neighbouring fragments.

[[Note added 2001:
since 1975 there have been huge developments in techniques for 'constraint propagation', including both hard and soft constraints. **]]**

But the input to Waltz' program was a representation of a perfectly connected and noise-free line drawing. Coping with disconnected images which have more defects, requires more prior knowledge about the structure of images and scenes depicted, and more sophisticated computational mechanisms.

Which dots in Figure 1 should be grouped into collinear fragments? By looking closely at the picture, you should be able to discern many more collinear groups than you previously noticed. That is, there are some lines which 'stand out' and are used in building an interpretation of the picture, whereas others for which the picture contains evidence are not normally noticed. Once you have noticed that a certain line 'stands out', it is easy to look along it picking out all the dots which belong to it, even though some of them may be 'attracted' by other lines too.

But how do you decide which lines stand out without first noticing all the collinear groups of dots? Are all the collinear dot-strips noticed unconsciously? What does that mean? Is this any different from unconsciously noticing grammatical relationships which make a sentence intelligible?

When pictures are made up of large numbers of disconnected and untidy fragments, then the interpretation problem is compounded by the problem of deciding which fragments to link together to form larger significant wholes. This is the 'segmentation' or 'agglomeration' problem. As so often happens in the study of mental processes, we find a circularity: once a fragment has been interpreted this helps to determine the others with which it should be linked, and once appropriate links have been set up the larger fragment so formed becomes less ambiguous and easier to interpret. It can then function as a recognisable cue. (The same circularity is relevant to understanding speech.)

9.7. The POPEYE program [1]

Our computer program breaks out of this circularity by sampling parts of the image until it detects a number of unambiguous fragments suggesting the presence of lines. It can then use global comparisons between different lines to see which are supported most strongly by relatively unambiguous fragments. These hypothesised bold lines then direct closer examination of their neighbourhoods to find evidence for bar-projections. Evidence which would be inconclusive out of context becomes significant in the context of a nearby bold line hypothesised as the edge of a bar an example of a 'Gestalt' directing the interpretation of details.

Thus, by using the fact that *some* fragments are fairly unambiguous, we get the process started. By using the fact that long stretches of relatively unambiguous fragments are unlikely to be spurious, the program can control further analysis and interpretations. Parallel pairs of bold lines are used as evidence for the presence of a bar. Many of the strategies used are highly fallible. They depend on assumption that the program inhabits a 'cognitively friendly' world, that is, that it will not be asked to interpret very messy, very confusing pictures. If it is, then, like people, it will become confused and start floundering.

Clusters of bar-like fragments found in this way can act as cues to generate further higher-level hypotheses, for example, letter hypotheses, which in turn control the interpretation of further ambiguous fragments. (For more details, see Sloman and Hardy 'Giving a computer gestalt experiences' and Sloman *et al.* 1978.) In order to give a program a more complete understanding of *our* concepts, we would need to embody it in a system that was able to move about in space and manipulate physical objects, as people do. This sort of thing is being done in other artificial intelligence research centres. However, there are still many unsolved problems. It will be a long time before the perceptual and physical skills of even a very young child can be simulated.

The general method of using relatively unambiguous fragments to activate prior knowledge which then directs attention fruitfully at more ambiguous fragments, seems to be required at all levels in a visual system. It is sometimes called the 'cue-schema' method, and seems to be frequently re-invented.

However, it raises serious problems, such as: how should an intelligent mechanism decide which schemas are worth storing in the first place, and how should it, when confronted with some cue, find the *relevant* knowledge in a huge memory store? (Compare chapter 8.) A variety of sophisticated indexing strategies may be required for the latter purpose. Another important problem is how to control the invocation of schemas when the picture includes cues for many different schemas.

Our program uses knowledge about many different kinds of objects and relationships, and runs several different sorts of processes in parallel, so that 'high-level' processes and (relatively) low-level' processes can help one another resolve ambiguities and reduce the amount of searching for consistent interpretations. It is also possible to suspend processes which are no longer useful, for example low-level analysis processes, looking for evidence of lines, may be terminated prematurely if some higher-level process has decided that enough has been learnt about the image to generate a useful interpretation.

This corresponds to the fact that we may recognise a whole (e.g. a word) without taking in all its parts. It is rational for an intelligent agent to organise things this way in a rapidly changing world where the ability to take quick decisions may be a matter of life and death.

Like people, the program can notice words and letters emerging out of the mess in pictures like Figure 1. As Kant says, the program has to work up the raw material by comparing representations, combining them, separating them, classifying them, describing their relationships, and so on. What Kant failed to do was describe such processes in detail.

9.8. The program's knowledge

In dealing with Figure 1 the program needs to know about several different domains of possible structures, depicted in Figure 3:

The domains of knowledge involved include:

a) The domain of 2-dimensional configurations of dots in a discrete rectangular array (the "dotty picture" domain).

b) The domain of 2-dimensional configurations of line-segments in a continuous plane. The configurations in the dotty picture domain *represent* configurations of such lines -- notice the differences between a collection of dots *being* a line segment, *lying on* a line segment and *representing* a line segment.

c) The (two-and-a-half-dimensional) domain of overlapping laminas composed of 'bars'. Patterns in the line-domain

d) represent configurations of bars and laminas made of rectangular bars.

e) An abstract domain containing configurations of 'strokes' which have orientations, lengths, junctions, and so on, analogous to lines, but with looser criteria for identity. Letters form a subset of this domain. Configurations in this domain are represented by configurations of laminas. That is, a bar-shaped lamina represents a possible stroke in a letter, but strokes of letters can also be depicted by quite different patterns (as in this printed text) which is why I say their domain is 'abstract' following Clowes, 1971.

f) An abstract domain consisting of sequences of letters. Known words form a subset of this domain.

Figure 3, below, illustrates some of the possible contents of the 2-D Domains used by the Popeye program in interpreting images like Figure 1.

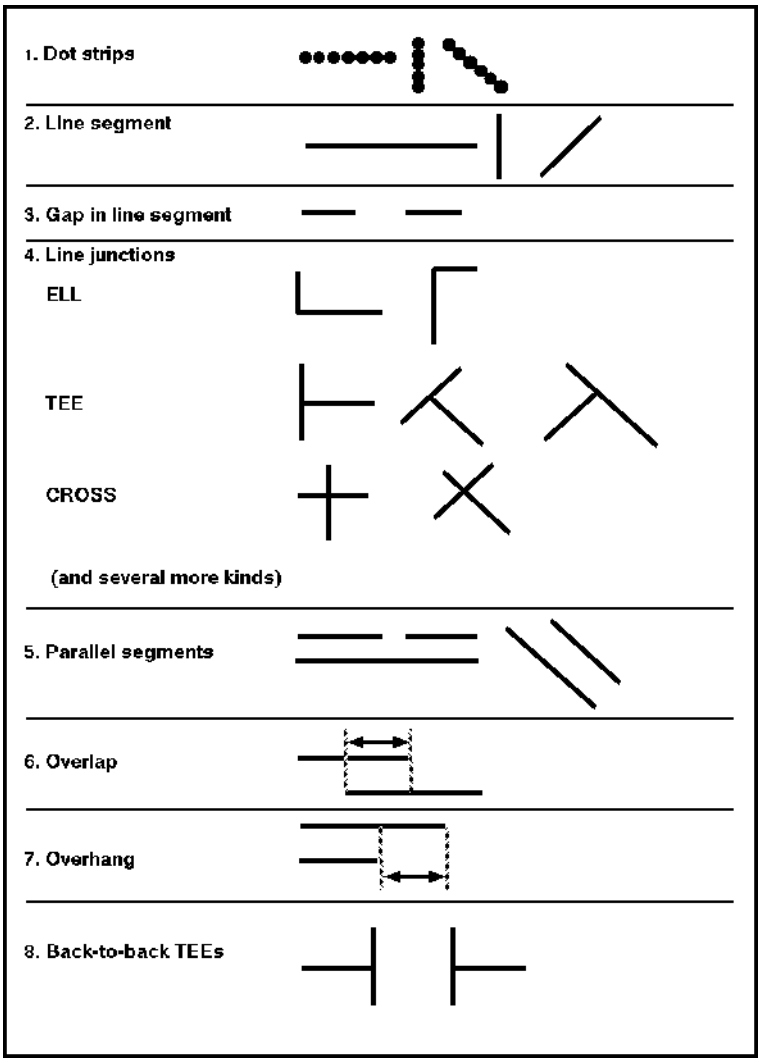


Figure 3

Some concepts relevant to the domain of 2 dimensional configurations of line-segments required for the interpretation of Figure 1. In this 2-D domain, nothing can be invisible or partly covered, unlike the domain of overlapping rectangular laminas shown in Figure 4, below. The process of interpreting Figure 1 includes identifying items in the 2-D domain and mapping them to items in the 2.5D domain of laminas.

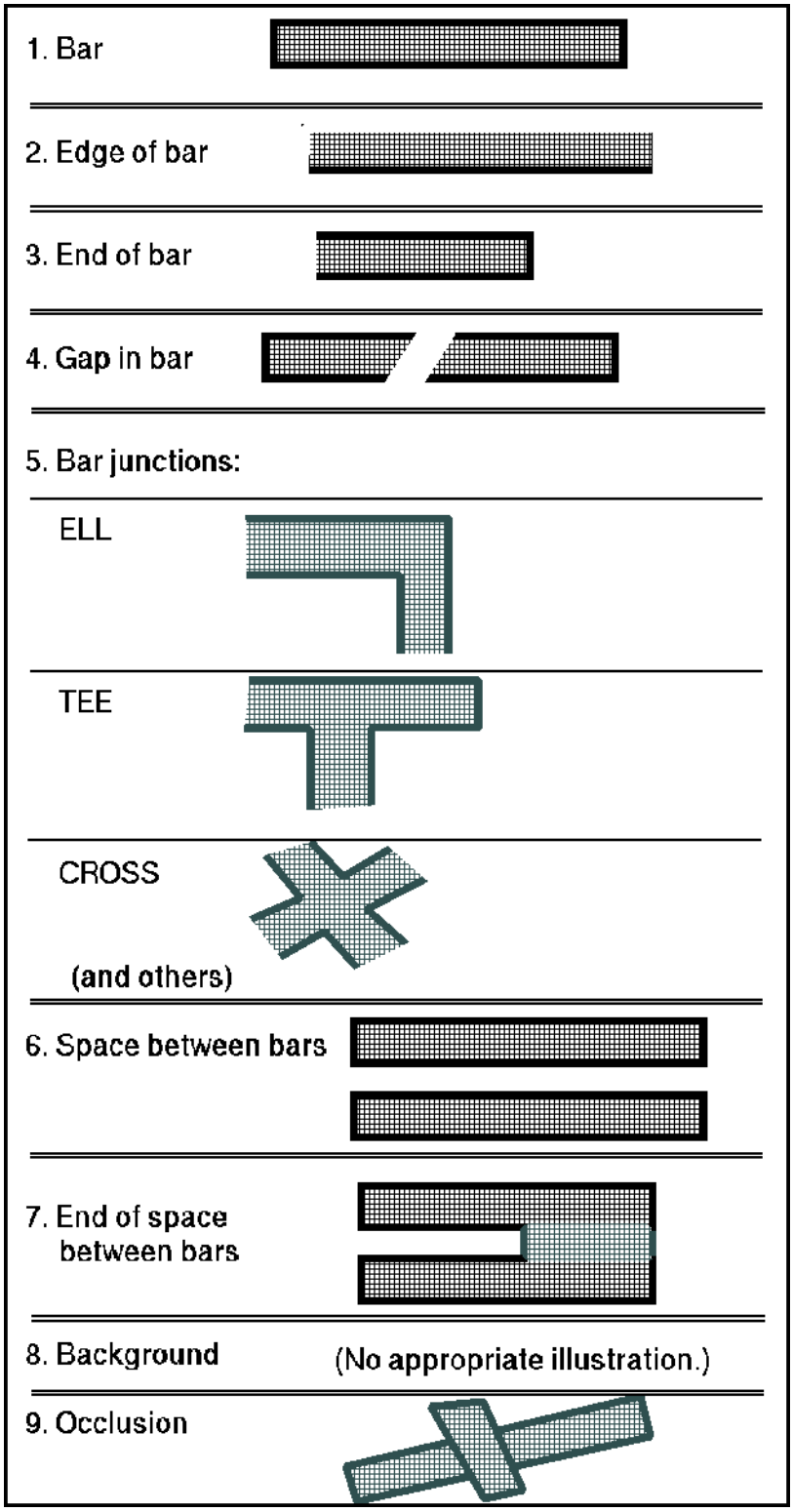
In particular the program has to know how to build and relate descriptions of structures in each of these domains, including fragments of structures. That is, the ability to solve problems about a domain requires an 'extension' of the domain to include possible fragments of well-formed objects in the domain Becker's 'phrasal lexicon' again. Our program uses many such intermediate concepts. Figures 3 and 4 list and illustrate some of the concepts relevant to the second and third domains. Figure 5 shows some of the cues that can help reduce the search for an interpretation. Figure 6 shows all the domains and some of the structural correspondences between items in those domains.

By making use of the notion of a series of domains, providing different 'layers' of interpretation, it is possible to contribute to the analysis of the concept of 'seeing as', which has puzzled some philosophers. Seeing X as Y is in general a matter of constructing a mapping between a structure in one domain and a possibly different structure in another domain. The mapping may use several intermediate layers.

[[Note added 2001:
our recent work on architectures containing a 'meta-management' layer suggests that *being aware of seeing X as Y* requires additional meta-management, i.e. self-monitoring processes, which are not essential for the basic processes of *seeing X as Y*, which could occur in simpler architectures, e.g. in animals that are not aware of their own mental processes (like most AI systems so far). **]]**

Figure 4 (Below)

Some concepts relevant to the domain of overlapping rectangular laminas. This sort of domain is sometimes described as "two and a half dimensional" (2.5D) because one object can be nearer or further than another, and because all or part of an object can be invisible because it is hidden behind another, unlike a purely 2D domain where everything is visible. Knowledge of such 2.5D concepts can help the search for a good interpretation of pictures like Figure 1. This raises problems about how the concepts are stored and indexed, how they are accessed by cues, and how ambiguities are resolved. Some of the discussion in Chapter 6 regarding special purpose and general purpose monitors is relevant.



Facts about one domain may help to solve problems about any of the others. For instance, lexical knowledge may lead to a guess that if the letters 'E', 'X' and T' have been found, with an unclear letter between them, then the unclear letter is 'I'. This in turn leads to the inference that there is a lamina depicting the 'I' in the scene. From that it follows that unoccluded edges of the lamina will be represented by lines in the hypothetical picture in domain (b). The inferred locations of these lines can lead to a hypothesis about which dots in the picture should be grouped together, and may even lead to

the conclusion that some dots which are not there should be there.

The program, like a person, needs to know that a horizontal line-segment in its visual image can represent (part of) the top or bottom edge of a bar, that an ELL junction between line segments can depict part of either a junction between two bars or a corner of a single bar. In the former case it may depict either a concave or a convex corner, and, as always, context will have to be used to decide which.

The program does not need to define concepts of one domain in terms of concepts from another. Rather the different domains are defined by their own primitive concepts and relations. The notion of 'being represented by' is not the same as the notion of 'being defined in terms of'. For instance, 'bar' is not defined in terms of actual and possible sense-data in the dot-picture domain, as some reductionist philosophical theories of perception would have us believe. Concepts from each domain are defined implicitly for the program in terms of structural relations and inference rules, including interpretation strategies.

So the organisation of the program is more consistent with a dualist or pluralist and wholistic metaphysics than with an idealist or phenomenalist reduction of the external world to sense-data, or any form of philosophical atomism, such as Russell and Wittgenstein once espoused.

Programs, like people, can in principle work out lots of things for themselves, instead of having them all programmed explicitly. For instance Figure 5 shows typical line-picture fragments which can be generated by laminas occluding one another. A program could build up a catalogue of such things for itself for instance by examining lots of drawings. Research is in progress on the problem of designing systems which learn visual concepts, possibly with the help of a teacher who chooses examples for the system to work on. (For example, see Winston, 1975.) It is certain that there are many more ways of doing such things than we have been able to think of so far. So we are in no position to make claims about which gives the best theory of how people learn.

[[Note added 2001:
In the decades since this book was written many more learning methods have been developed for vision and other aspects of intelligence, though surprisingly few of them seem to involve the ability to learn about different classes of structures in domains linked by representation relationships. Many of them attempt to deal with fairly direct mappings between configurations detectable in image sequences and abstract concepts like "person walking". For examples see journals and conference proceedings on machine vision, pattern recognition, and machine learning.**]]**

Currently our program starts with knowledge which has been given it by people (just as people have to start with knowledge acquired through a lengthy process of biological evolution). Perhaps, one day, some of the knowledge will be acquired by a machine itself, interacting with the world, if a television camera and mechanical arm are connected to the computer, as is already done in some A.I. research laboratories. However, real learning requires much more sophisticated programs than programs which have a fixed collection of built-in abilities. (Some of the problems of learning were discussed previously in Chapter 6 and Chapter 8.)

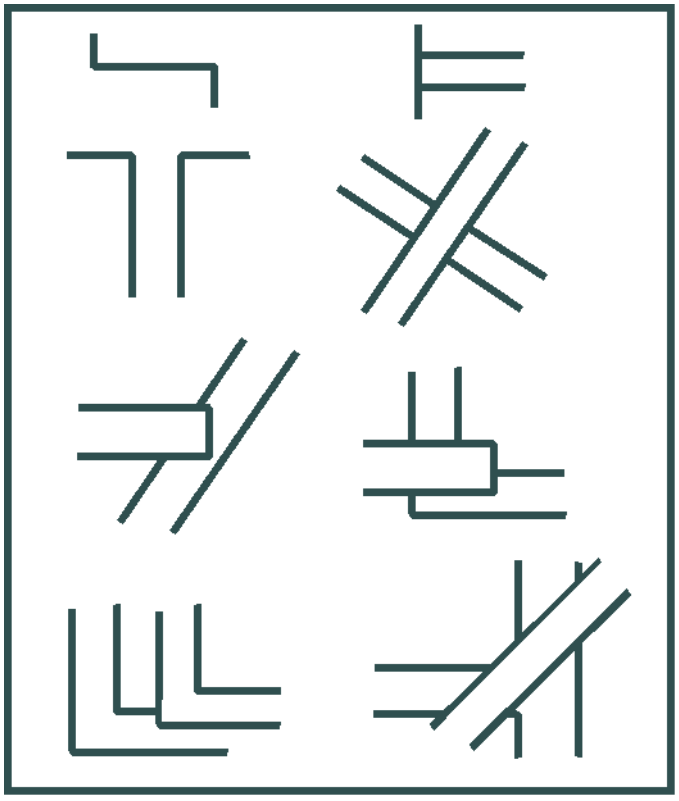


Figure 5

This shows a number of sub-configurations within the 2-D line-segment domain of Figure 3 which are likely to occur in images depicting overlapping laminas from the domain of Figure 4. A set of 2-D line images depicting a different class of laminas, or depicting objects in a different domain, e.g. 3-D forest scenes, would be likely to include a different class of sub-configurations made of lines.

Likewise in depictions of forest scenes, commonly occurring configurations in the dotty picture domain would be different from those found in Figure 1.

Knowledge of commonly occurring sub-structures in images, corresponding to particular domains represented, like knowledge about the objects represented, can help the interpretation process. This is analogous to processes in language-understanding in which knowledge of familiar phrases is combined with knowledge of a general grammar which subsumes those phrases. (Becker 1975)
[[This caption was substantially extended in 2001]]

Given structural definitions of letters, and knowledge of the relations between the different domains illustrated in Figure 6, a program might be able to work out or learn from experience that certain kinds of bar junctions (Figure 4), or the corresponding 2-D line configurations (Figures 3 and 5), occur only in a few of them, and thus are useful disambiguating cues. This will not be true of all the fragments visible in Figure 1. Thus many fragments will not be recognised as familiar, and spurious linkages and hypotheses will therefore not be generated. If the program were familiar with a different world, in which other fragments were significant, then it might he more easily be confused by Figure 1. So additional knowledge is not always helpful. (Many works of art seem to require such interactions between different domains of knowledge.)

A program should also be able to 'learn' that certain kinds of fragments do not occur in any known letter, so that if they seem to emerge at any stage this will indicate that picture fragments have been wrongly linked together. This helps to eliminate fruitless searches for possible interpretations. So the discovery of anomalies and impossibilities may play an important role in the development of rational behaviour. A still more elaborate kind of learning would involve discovering that whether a fragment is illegitimate depends on the context. Fragments which are permissible within one alphabet may not be permissible in another. Thus the process of recognising letters is facilitated by knowledge of the alphabet involved, yet some letter recognition may be required for the type of alphabet to be inferred: another example of the kind of circularity, or mutual dependence, of sub-abilities in an intelligent system.

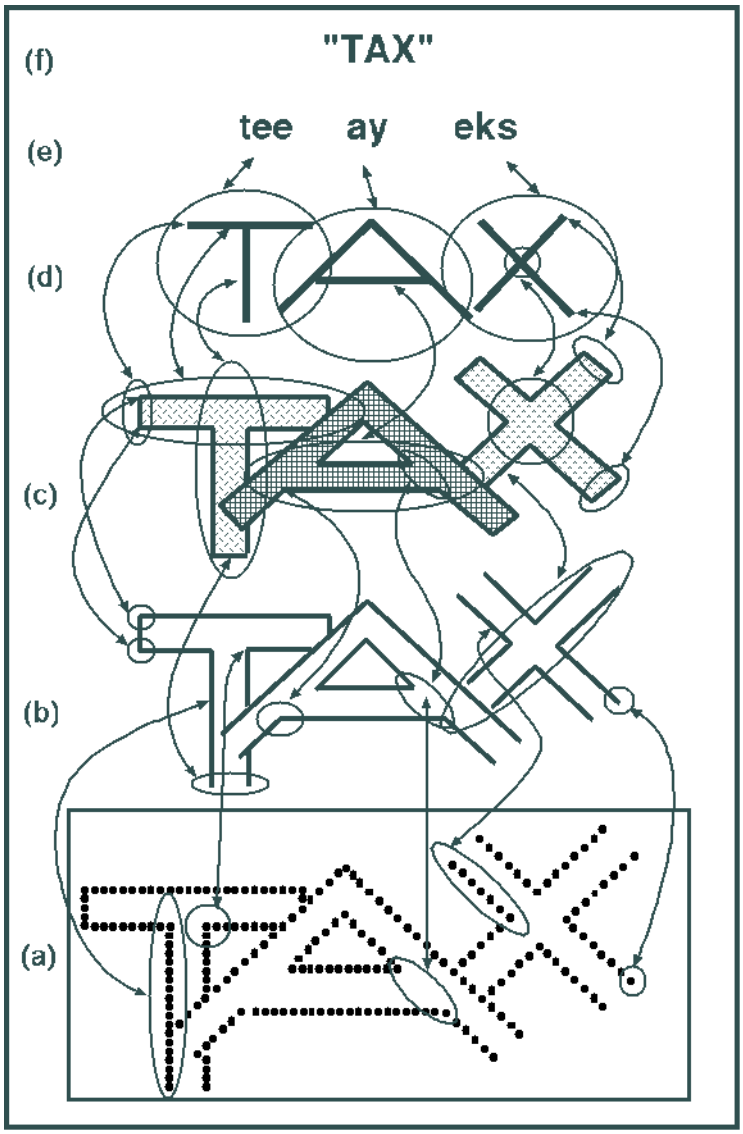


Figure 6

This shows how several layers of interpretation may be involved in seeing letters in a dot-picture. Each layer is a domain of possible configurations in which substructures may represent or be represented by features or substructures in other layers. The following domains are illustrated: (a) configurations of dots, spaces, dotstrips, etc., (b) configurations of 2-D line-segments, gaps, junctions, etc., (c) configurations of possibly overlapping laminas (plates) in a 2.5D domain containing bars,

bar-junctions, overlaps, edges of bars, ends of bars, etc., (d) a domain of stroke configurations where substructures can represent letters in a particular type of font, (e) a domain of letter sequences, (f) a domain of words composed of letter sequences.

NOTE [13 Jan 2007; Clarified 1 Jul 2015]:
The original diagram in Figure 6 suggested that all information flows upwards. That is not how the program worked: there was a mixture of bottom-up, top-down and middle-out processing, and the original arrows in the figure showing information flow have been replaced with bi-directional arrows to indicate this.

9.10. Style and other global features

Knowledge of ’picture styles’ can also play an important role in the process of perception and interpretation. Variations in style include such things as whether the letters are all of the same height and orientation, whether the bars are all of the same width, whether the letters in words tend to be jumbled, or overlapping, or at stepped heights, and so on. Notice that some of these stylistic concepts depend on quite complex geometrical relationships (for instance, what does ’jumbled’ mean?). If the program can take note of clues to the style of a particular picture, during its analysis, this can help with subsequent decisions about linking or interpreting fragments. If you know the sizes of letters, for instance, then you can more easily decide whether a line segment has a bit missing.

Hypotheses about style must, of course, be used with caution, since individual parts of a picture need not conform to the overall style. Local picture evidence can over-ride global strategies based on the inferred style provided that the program can operate in a mode in which it watches out for evidence conflicting with some of its general current assumptions, using monitors of the sorts described in Chapter 6.

9.11. Perception involves multiple co-operating processes

Our program includes mechanisms which make it possible to set a number of different processes going in parallel, for example, some collecting global statistics about the current picture, some sampling the picture for dot-configurations which might represent fragments of lines, others keeping track of junctions between lines, or attempting to interpret parallel segments as bars, some trying to interpret bars as strokes of letters, and so on.

This parallelism is required partly because, with a large amount of information available for analysis and interpretation, it may not be easy to decide what to do next, for example, which configurations to look for in the picture, and where to look for them. Deciding between such alternatives itself requires analysis and interpretation of evidence and at first it will not be obvious where the important clues are, nor what they are. So initially many on-going processes are allowed to coexist, until items both unambiguous and relatively important emerge, such as a long line, an unambiguous clue to the location of a bar, some aspect of the style, or a set of linked bar fragments which uniquely identify a letter.

When fragments forming clear-cut cues emerge, they can invoke a ’higher-level’ schema which takes control of processing for a while, interrupting the ’blind’ searching for evidence, by directing attention to suitable parts of the picture and relevant questions.

If higher level processes form a plausible hypothesis, this may suppress further analysis of details by lower level processes. For instance, recognition of fragments of ’E’, or ’X’, and of "I", where there appear to be only about four letters, might cause a program (or person) to jump to the conclusion that the word is ’EXIT’, and if this fits into the context, further examination of lines to check out on

remaining strokes of letters, and the missing 1', might then be abandoned. This ability to jump to conclusions on the basis of partial analysis may be essential to coping with a rapidly changing world. However it depends on the existence of a fair amount of redundancy in the sensory data: that is, it assumes a relatively 'friendly' (in the sense defined previously) world. It also requires an architecture able to support multiple concurrent processes and the ability for some of them to be aborted by others when their activities are no longer needed.

This type of programming involves viewing perception as the outcome of very large numbers of interacting processes of analysis, comparison, synthesis, interpretation, and hypothesis-testing, most, if not all, unconscious. On this view the introspective certainty that perception and recognition are 'direct', 'unmediated' and involve no analysis, is merely a delusion. (This point is elaborated in the papers by Weir -- see Bibliography.)

This schizophrenic view of the human mind raises in a new context the old problem: what do we mean by saying that consciousness is 'unitary' or that a person has one mind? The computational approach to this problem is to ask: how can processes be so related that all the myriad sub-tasks may be sensibly co-ordinated under the control of a single goal, for instance the goal of finding the word in a spotty picture, or a robot's goal of using sensory information from a camera to guide it as it walks across a room to pick up a spanner? See also chapter 6 and chapter 10.

[[Note added 2001:
At the time the program was being developed, we had some difficulty communicating our ideas about the importance of parallel processing concerned with different domains because AI researchers tended to assume we were merely repeating the well-known points made in the early 1970s by Winograd, Guzman and others in the MIT AI Lab, about "heterarchic" as opposed to "hierarchic" processing.
Heterarchic systems, dealt, as ours did, with different domains of structures and relations between them (e.g. Winograd's PhD thesis dealt with morphology, syntax, semantics and a domain of three dimensional objects on a table).
Both models involve mixtures of data-driven (bottom-up) and hypothesis-driven (top-down) processes.
Both allow interleaving of processes dealing with the different domains -- unlike *hierarchic* or *pass-oriented* mechanisms which first attempt to complete processing in one domain then pass the results to mechanisms dealing with another domain, as in a processing pipeline.

The main differences between heterarchy and our model were as follows:

a) In an implementation of "heterarchic" processing there is typically only *one* locus of control at any time. Thus processing might be going on in a low level sub-system or in a high level sub-system, but not both in parallel with information flowing between them.

b) In those systems decisions to transfer control between sub-systems were all taken explicitly by processes that decided they needed information from another system: e.g. a syntactic analyser could decide to invoke a semantic procedure to help with syntactic disambiguation, and a semantic procedure could invoke a syntactic analyser to suggest alternative parses.

c) In that sort of heterarchic system it is not possible for a process working in D1 to be *interrupted* by the arrival of new information relevant to the current sub-task, derived from processing in D2.

d) Consequently, if a process in that sort heterarchic system gets stuck in a blind-alley and does not notice this fact it may remain stuck forever.

The POPEYE architecture was designed to overcome these restrictions by allowing processing to occur concurrently in different domains with priority mechanisms in different domains determining which sub-processes could dominate scarce resources. Priorities could change, and attention within a domain could therefore be switched, as a result of arrival of new information that was not explicitly asked for.

In this respect the POPEYE architecture had something in common with neural networks in which information flows between concurrently processing sub-systems (usually with simulated concurrency). Indeed, a neural net with suitable symbol-manipulating sub-systems could be used to implement something like the POPEYE architecture, though we never attempted to do this for the whole system. After this chapter was written, work was done on implementing the top level word-recognizer in POPEYE as a neural net to which the partial results from lower level systems could be fed as they became available. **]]**

9.12. The relevance to human perception

The world of our program is *very* simple. There are no curves, no real depth, no movement, no forces. The program cannot act in this world, nor does it perceive other agents. Yet even for very simple worlds, a computer vision program requires a large and complex collection of knowledge and abilities. From such attempts to give computers even fragmentary human abilities we can begin to grasp the enormity of the task of describing and explaining the processes involved in *real* human perception. Galileo's relationship to the physics of the 1970s may be an appropriate and humbling comparison.

In the light of this new appreciation of the extent of our ignorance about perceptual processes, we can see that much philosophical discussion hitherto, in epistemology, philosophy of mind, and aesthetics, has been based on enormous over-simplifications. With hindsight much of what philosophers have written about perception seems shallow and lacking in explanatory power. But perhaps it was a necessary part of the process of cultural evolution which led us to our present standpoint.

Another consequence of delving into attempts to give computers even very simple abilities is that one acquires enormous respect for the achievements of very young children, many other animals, and even insects. How does a bee manage to . land on a flower without crashing into it?

Many different aspects of perception are being investigated in artificial intelligence laboratories. Programs are being written or have been written which analyse and interpret the following sorts of pictures or images, which people cope with easily.

a) Cartoon drawings.

b) Line drawings of three dimensional scenes containing objects with straight edges, like blocks and pyramids.

c) Photographs or television input from three-dimensional scenes, including pictures of curved objects.

d) Stereo pairs from which fairly accurate depth information can be obtained.

e) Sequences of pictures representing moving objects, or even television input showing moving objects.

f) Satellite photographs, which give geological, meteorological, or military information. (Unfortunately, some people are unable to procure research funds unless they pretend that their work is useful for military purposes and, even more unfortunately, it sometimes is.)

g) Pictures which represent 'impossible objects', like Escher's drawings. Like people, a program may be able to detect the impossibility (see Clowes, 1971, Huffman, 1971, and Draper (to appear)).

Some of the programs are in systems which control the actions of artificial arms, or the movements of vehicles. The best way to keep up with this work is to read journal articles, conference reports, and privately circulated departmental reports. Text-books rapidly grow out of date. (This would not be so much of a problem if we all communicated via a network of computers and dispensed with books! But that will not come for some time.)

Each of the programs tackles only a tiny fragment of what people and animals can do. For example, the more complex the world the program deals with the less of its visible structure is perceived and used by the program. The POPEYE program deals with a very simple world because we wanted it to have a fairly full grasp of its structure (though even that is proving harder than we anticipated). One of the major obstacles to progress at present is the small number of memory locations existing computers contain, compared with the human brain. But a more important obstacle is the difficulty of articulating and codifying all the different kinds of structural and procedural knowledge required for effective visual perception. There is no reason to assume that these obstacles are insuperable in principle, though it is important not to make extravagant claims about work done so far. For example, I do not believe that the progress of computer vision work by the end of this century will be adequate for the design of domestic robots, able to do household chores like washing dishes, changing nappies on babies, mopping up spilt milk, etc. So, for some time to come we shall be dependent on simpler, much more specialised machines.

9.13. Limitations of such models

It would be very rash to claim that POPEYE, or any other existing artificial intelligence program, should be taken seriously as a theory explaining human abilities. The reasons for saying that existing computer models cannot be accepted as explaining how people do things include:

a) People perform the tasks in a manner which is far more sensitive to context, including ulterior motives, emotional states, degree of interest, physical exhaustion, and social interactions. Context may affect detailed strategies employed, number of errors made, kinds of errors made, speed of performance, etc.

b) People are much more flexible and imaginative in coping with difficulties produced by novel combinations, noise, distortions, missing fragments, etc. and at noticing short cuts and unexpected solutions to sub-problems.

c) People learn much more from their experiences.

d) People can use each individual ability for a wider variety of purposes: for instance we can use our ability to perceive the structure in a picture like Figure 1 to answer questions about spaces between the letters, to visualise the effects of possible movements, to colour in the letters with different paints, or to make cardboard cut-out copies. We can also interpret the dots in ways which have nothing to do with letters, for instance seeing them as depicting a road map.

e) More generally, the mental processes in people are put to a very wide range of *practical* uses, including negotiating the physical world, interacting with other individuals, and fitting into a society. No existing program or robot comes anywhere near matching this.

These discrepancies are not directly attributable to the fact that computers are not made of neurons, or that they function in an essentially serial or digital fashion, or that they do not have biological origins. Rather they arise mainly from huge differences in the amount and organisation of practical and theoretical knowledge, and the presence in people of a whole variety of computational processes to do with motives and emotions which have so far hardly been explored.

A favourite game among philosophers and some 'humanistic' psychologists is to list things computers cannot do. (See the book by Dreyfus for a splendid example.) However, any sensible worker in artificial intelligence will also spend a significant amount of time listing things computers cannot do yet! The difference is that the one is expressing a prejudice about the limitations of computers, whereas the other (although equally prejudiced in the other direction, perhaps) is doing something more constructive: trying to find out exactly what it is about existing programs that prevents them doing such things, with a view to trying to extend and improve them. This is more constructive because it leads to advances in computing, and it also leads to a deeper analysis of the human and animal abilities under investigation.

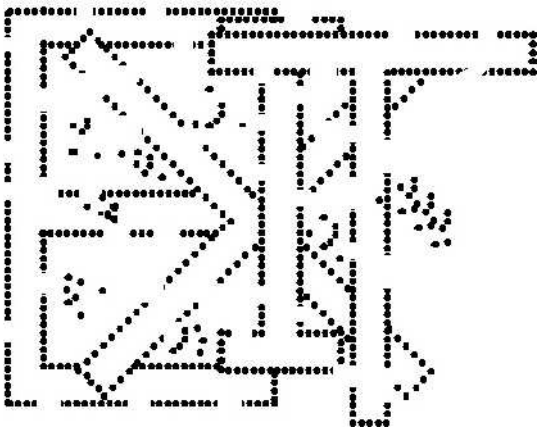
As suggested previously in Chapter 5, attempting to *prove* that computers cannot do this or that is a pointless exercise since the range of abilities of computers, programming languages and programs is constantly being extended, and nobody has any formal characterisation of the nature of that process which could serve as a basis for establishing its limits. The incompleteness and unsolvability theorems of Goedel and others refer only to limitations of narrowly restricted *closed* systems, which are quite unlike both people and artificial intelligence programs which communicate with the world.

This chapter has presented a few fragments from the large and growing collection of ideas and problems arising out of A.I. work on vision. I have begun to indicate some of the connections with philosophical issues, but there is a lot more to be said. The next.chapter develops some of the points of contact at greater length.

Endnotes

(1) The name 'POPEYE' comes from the fact that the program is written in POP-2, a programming language developed at Edinburgh University for artificial intelligence research. See Burstall *et al.* A full account of how POPEYE works, with an analysis of the design problems could fill a small book. This chapter gives a superficial outline, focusing on aspects that are relevant to a general class of visual systems. Details will be published later. The work is being done with David Owen, Geoffrey Hinton, and Frank O'Gorman. Paul (1976) reports some closely related work.

[[**Note added 1 Jul 2015** The published version of this book did not make it clear that the POPEYE program was able to cope with pictures similar to Figure 1 but with additional positive and negative noise added, as illustrated here:



A consequence of the parallelism and the bi-directionality of information flow was that the program could often conclude that the word, or some higher level structure, had been identified before all processing of the evidence had been completed. Sometimes that identification was mistaken e.g. because the addition of positive and negative noise, or overlap of letters, had obscured some of the evidence, and further processing would reveal the error. This seems to reflect the fact that humans sometimes think they have recognized someone or something (and may then greet the person) and soon after that realise, with the person out of sight, that the recognition was mistaken, presumably because more of the links relating data and interpretation fragments have been computed. This familiar feature of human vision, and related errors of proof-reading text, were among the motivations for the design of Popeye.]]

[[**Notes added Sept 2001.**

(a) A more complete description of Popeye was never published and the application for a research grant to extend the project around 1978 was unsuccessful. Both appear in part to have been a consequence of the view then gaining currency, based largely on the work of David Marr, that AI vision researchers who concentrated on mixtures of top-down and bottom-up processes were deluded, usually because they were misled by problems arising from the use of *artificial* images.

Marr's ideas about mistakes in AI vision research were originally published in MIT technical reports that were widely circulated in the mid 1970s. He died, tragically, in 1981, and the following year his *magnum opus* was published: D. Marr, *Vision*, 1982, Freeman, 1982.

(b) Marr's criticism of AI vision research was based in part on the claim that natural images are far richer in information and if only visual systems took account of that information they would not need such sophisticated bi-directional processing architectures. My own riposte at the time (also made by some other researchers) was:

- On the one hand human vision can cope very well with these artificial and degraded images, e.g. in cartoon drawings, so there is a fact to be explained and modelled. Moreover that ability to deal effortlessly with cartoon drawings may have some deep connection with intermediate stages of processing in natural perception.
- In addition even natural images are often seriously degraded -- by poor light, dirty windows, mist, dust-storms, occluding foliage, rapid motion, other features of the environment, and damage to eyes.

(c) In the late 1970s there was also growing support for a view also inspired in part by Marr's work, namely, that symbol manipulating mechanisms and processes of the sorts described in this chapter and elsewhere in this book were not really necessary, as everything could be achieved by emergent features of collections of 'local cooperating processes' such as neural nets.

Neural nets became increasingly popular in the following years, and they have had many successful applications, though it is not clear that their achievements have matched the expectations of their proponents. Work on neural nets and other learning or self-organising systems, including the more recent work on evolutionary computation, is often (though not always) driven by a desire to avoid the need to understand a problem and design a solution: the hope is that some *automatic* method will make the labour unnecessary. My own experience suggests that until people have actually solved some of these problems themselves they will not know what sort of learning mechanism or self-organising system is capable of

solving them. However, when we have done the analysis required to design the appropriate specialised learning mechanisms we may nevertheless find that the products of such mechanisms are beyond our comprehension. E.g. the visual ontology induced by a self-organising perceptual system that we have designed may be incomprehensible to us.

What I am criticising is not the search for learning systems, or self-organising systems, but the search for *general-purpose* automatic learning mechanisms equally applicable to all sorts of problems. Different domains require different sorts of learning processes, e.g. learning to walk, learning to see, learning to read text, learning to read music, learning to talk, learning a first language, learning a second language, learning arithmetic, learning meta-mathematics, learning quantum mechanics, learning to play the violin, learning to do ballet, etc. In some cases the learning requires a specific *architecture* to be set up within which the learning can occur. In some cases specific forms of representation are required, and mechanisms for manipulating them. In some cases specific forms of interaction with the environment are required for checking out partial learning and driving further learning. And so on.

(d) At the time when the Popeye project was cancelled for lack of funds, work was in progress to add a neural net-like subsystem to help with the higher levels of recognition in our pictures of jumbled letters. I.e. after several layers of interpretation had been operating on an image like Figure 1, a hypothesis might begin to emerge concerning the letter sequence in the second domain from the top. In the original Popeye program a technique analogous to spelling correction was used to find likely candidates and order them, which could, in turn, trigger top-down influences to check out specific ambiguities or look for confirming evidence. This spelling checker mechanism was replaced by a neural net which could be trained on a collection of known words and then take a half-baked letter sequence and suggest the most likely word. (This work was done by Geoffrey Hinton, who was then a member of the Popeye project, and later went on to be one of the leaders in the field of neural nets.)

(e) Despite the excellence of much of Marr's research (e.g. on the cerebellum) I believe that AI research on vision was dealt a serious body blow by the publication of his views, along with the fast growing popularity of neural nets designed to work independently of more conventional AI mechanisms, and likewise later work on statistical or self-organising systems, motivated in part by the vain hope that by writing programs that learn for themselves or evolve automatically, we can avoid the need to understand, design and implement complex visual architectures like those produced by millions of years of evolution.

Certainly no matter what kinds of high level percept a multi-layer interpretation system of the sort described in this chapter produces, it is possible to mimic some of its behaviour by using probabilistic or statistical mechanism to discover correlations between low level input configurations and the high level descriptions. This is particularly easy where the scenes involve isolated objects, or very few objects, with not much variation in the arrangements of objects, and little or no occlusion of one object by another.

The problem is that in real life, including many practical applications, input images very often depict cluttered scenes with a wide variety of possible objects in a wide variety of possible configurations. If the image projection and interpretation process involves several intermediate layers, as in figure 6 above, each with a rich variety of permitted structures, and complex structural relations between the layers, the combinatorics of the mapping between input images and high level percepts can become completely intractable, especially if motion is also allowed and some objects are flexible. One way of achieving tractability is to decompose the problem into tractable sub-problems whose solutions can interact possibly aided by background knowledge. This seems to me to require going back to some of the approaches to vision that were being pursued in the 1970s including approaches involving the construction and analysis of *structural descriptions* of intermediate configurations. The computer power available for this research in the 1970s was a major factor in limiting success of that approach: if it takes 20 minutes simply to find the edges in an image of a cup and saucer there are strong pressures to find short cuts, even if they don't generalise.

(f) The growing concern in the late 1970s and early 1980s for *efficiency*, discouraged the use of powerful AI programming languages like Lisp and Pop-11, and encouraged the use of lower level batch-compiled languages like Pascal and C and later C++. These languages were not as good as AI languages for expressing complex operations involving structural descriptions, pattern matching and searching, especially without automatic garbage collection facilities. They are also not nearly as flexible in permitting task-specific syntactic extensions as AI languages, which allow the features of different problems to be expressed in different formalisms within the same larger program. Moreover AI languages with interpreters or incremental compilers provide far better support support for interactive exploration of complex domains where the algorithms and representations required cannot be specified in advance of the programming effort, and where obscure conceptual bugs often require interactive exploration of a running system.

However, the emphasis on *efficiency* and *portability* pressurised researchers to use the non-AI languages, and this subtly pushed them into focusing on problems that their tools could handle, alas.

Robin Popplestone (the original inventor of Pop2) once said to me that he thought the rise in popularity of C had killed off research in the real problems of vision. That may be a slight exaggeration.

(g) For a counter example to the above developments see Shimon Ullman, *High-level vision: Object recognition and visual cognition*, MIT Press, 1996. I have the impression that there may now be a growing collection of AI vision researchers who are dissatisfied with the narrow focus and limited applicability of many machine vision projects, and would welcome a move back to the more ambitious earlier projects, building on what has been learnt in recent years where appropriate. This impression was reinforced by comments made to me by several researchers at the September 2001 conference of the British Machine Vision Association.

(h) Besides the obvious limitations due to use of artificially generated images with only binary pixel values, there were many serious limitations in the Popeye project, including the restriction to objects with straight edges, the lack of any motion perception, and the lack of any perception of 3-D structure and relationships (apart from the partial depth ordering in the 2-D lamina domain). Our defence against the criticism of over-simplification was that we thought some of the architectural issues relevant to processing more complex images or image sequences, dealing with more complex environments, could usefully be addressed in an exploration of our artificial domain, if only by producing a "proof of principle", demonstrating how cooperative processes dealing with different domains could cooperate to produce an interpretation without time-consuming search.

(i) In the 20 years following the Popeye project (and this book) I gradually became aware of more serious, flaws, as follows.

- I had assumed that although seeing involved processing structures in different domains in parallel, it was necessarily a unitary process in that all those processes contributed to the same eventual high level task of acquiring information about the structure and contents of the environment. Later it became clear that this was a mistake: there are different architectural layers using visual information in parallel for quite different purposes, e.g. posture control, planning ahead of actions to be performed, fine-control of current actions through feedback loops, answering questions about how something works, social perception, and so on. The different sub-mechanisms require different information about the environment, which they can acquire in parallel, often sharing the same low level sensors.

  Some of these are evolutionarily very old mechanisms shared with many animals. Others use much newer architectural layers, and possibly functions and mechanisms unique to humans.

  This point was already implicit in my discussion of the overall architecture with its multiple functions in Chapter 6, e.g. in connection with monitors.

- At that time I shared the general view of AI researchers and many psychologists that the primary function of perception, including vision, was to provide information about the environment in the form of some sort of "declarative" *description* or *information structure* that could be used in different ways in different contexts. Later I realised that another major function of perceptual systems was to trigger appropriate *actions* directly, in response to detected patterns.

  Some of these responses were external and some internal, e.g. blinking, saccadic eye movements, posture control, and some internal emotional changes such apprehension, sexual interest, curiosity, etc.

  This use of perceptual systems seems to be important both in innate reflexes and in many learnt skills for instance athletic skills.

  Of course, when I started work on this project I already knew about reflexes and trained high speed responses, as did everyone else: I simply did not see their significance for a visual architecture (though I had read J.J.Gibson's book *The senses considered as perceptual systems*, which made the point.)

  Later this idea became central to development of the theory about a multi-layer architecture, mentioned above, in which reactive and deliberative processes run in parallel often starting from the same sensory input. This theme is still being developed in papers in the Cogaff project.

- Like many researchers on vision in AI and psychology, I had assumed that insofar as vision provided factual information about the environment it was information about *what exists* in the environment. Later I realised that what is equally or more important, is *awareness of what might exist, and the constraints* on what might exist, e.g. "that lever can rotate about that point, though the rotation will be stopped after about 60 degrees when the lever hits the edge of the frame".

  The need to see what is and is not possible, in addition to what is actually there, has profound implications for the types of information representations used within the visual system: structural descriptions will not suffice. Several papers on this are included in the Cogaff web site, some mentioned below.

The last critique was inspired by J.J.Gibson's notion of "affordance". See for example his book, *The Ecological Approach to Visual Perception* originally published in 1979. Although I rejected some of his theories (e.g. the theory that perception could somehow be direct, and representation free) the theory that vision was about detecting affordances seemed very important. I.e. much of what vision (and perception in general) is about is not just provision of information about what is *actually* in the environment, but, more importantly, information about what sorts of things are *possible* in a particular environment that might be useful or harmful to the viewer, and what the *constraints* on such possibilities are.

Although I think very little progress has been made on this topic, several of my papers explored aspects of this idea, e.g.

- A. Sloman, 'Image interpretation: The way ahead?'
  Invited talk, in *Physical and Biological Processing of Images*, Editors: O.J.Braddick and A.C. Sleigh, Pages 380--401, Springer-Verlag, 1982.
- A. Sloman, 'On designing a visual system (Towards a Gibsonian computational model of vision)', in *Journal of Experimental and Theoretical AI*, 1, 4, pp. 289--337, 1989.
- A. Sloman, 'Actual Possibilities', in Eds. L.C. Aiello and S.C. Shapiro, *Principles of Knowledge Representation and Reasoning: Proceedings of the Fifth International Conference (KR '96)*, pp. 627--638, 1996,
- A. Sloman, 'Diagrams in the mind', in *Diagrammatic Representation and Reasoning*, Eds. M. Anderson, B. Meyer and P. Olivier, Springer-Verlag, 2001,
- A. Sloman 'Evolvable Biologically Plausible Visual Architectures', in *Proceedings British Machine Vision Conference*, Eds T.Cootes and C.Taylor. 2001.
- Talks/presentations on vision in http://www.cs.bham.ac.uk/research/projects/cogaff/talks/ and in http://www.cs.bham.ac.uk/research/projects/cosy/papers/.
- During work on the CoSy robotic project in 2005 I became increasingly aware that in addition to concurrent perception of *structures* at different levels of abstraction a human-like (or intelligent robot's) vision system would need to perceive *processes* of different sorts, and different levels of abstraction concurrently, as explained in this PDF presentation:
  A (Possibly) New Theory of Vision (2005).

The above papers are all available here http://www.cs.bham.ac.uk/research/cogaff/ along with additional papers on architectural layers and their implications for the evolution of visual systems and action systems.

(j) The Edinburgh AI language *Pop2* mentioned above later evolved into *Pop-11*, which became the core of the Poplog system developed at Sussex University and marketed for several years by ISL, who contributed further developments. It is now available free of charge with full system sources for a variety of platforms here:
http://www.cs.bham.ac.uk/research/poplog/freepoplog.html, including materials supporting teaching and research on vision, developed by David Young at Sussex University.
]]

Book contents page
Next: Chapter 10
Last updated: 4 Jun 2007; 19 Sep 2010; Re-formatted 1 Jul 2015