DRAPER-1

Using models to augment rule-based programs

Stephen W. Draper
Cognitive Studies Programme
University of Sussex
Brighton, Sussex, U.K.

Abstract

This paper discusses the design of a program that tackles the ambiguity resulting from the interpretation of line-drawings by means of geometric constraints alone. It does this by supplementing its basic geometric reasoning by means of a set of models of various sizes. Earlier programs are analysed in terms of models, and three different functions for models are distinguished. Finally, principles for selecting models for the present purpose are related to the concept of a "mapping event" between the picture and scene domains.

Introduction

The line-labelling scheme proposed by Huffman (1971) and Clowes (1971) in effect posed the problem of producing all the interpretations in terms of line-labels of a line-drawing that can be made on the assumption that it is a projection of a scene composed of polyhedra with trihedral vertices, plane surfaces, and straight edges, and also that the viewpoint does not give rise to any accidental alignments of vertices or edges (i.e. any "accidentals"). The line-labelling scheme offered only a partial solution, and left two major challenges for future program designs: to relax the non-accidental and trihedral restrictions (i.e.to allow accidental alignments and vertices with more or fewer than three surfaces), and to generate only those labellings that are geometrically consistent with the assumptions - Huffman himself showed that the line-labelling scheme was inadequate in this respect. Waltz' (1972) program did nothing towards the latter goal, and made only ad hoc attempts at the former by including a few hand-picked accidental and multihedral junction labellings (his real contribution was to the expression of knowledge about shadows). Mackworth's program Poly (Mackworth 1973) achieved the first and partly achieved the second goal, and work by the author on sidedness reasoning including a design for a program called Ellsid (Draper 1978, forthcoming) has completed this goal. It turns out, however, that the purely geometric constraints thus fully captured allow an enormous number of interpretations (many hundreds even for simple drawings of a cube or a tetrahedron) even though people see only one or two. Clearly the next task is to attempt to model the choice of interpretation made by us from among those geometrically possible.

Much of this ambiguity comes from allowing accidentals indiscriminately - in effect this treats each picture region as the projection of (part of) an opaque plate with no necessary point of contact with any other, and many of the interpretations represent such weird, disconnected scenes where the edges of floating plates are lined up in various unlikely ways. This does not however account for all the ambiguity since even excluding accidentals still allows numerous odd interpretations. It is now clear that the trihedral and accidental restrictions in the Huffman-Clowes scheme were responsible for keeping the interpretations produced in fairly close correspondence to human interpretations, even though their rigid application prevented the interpretation of some simple pictures. Kanade (1978) shows how even a slight relaxation of the restrictions (redefining the trihedral restriction to allow up to three surfaces at a vertex, which can be either laminae or faces bounding a solid volume) greatly multiplies the possible interpretations of simple pictures and that extra scene constraints must then be mobilised. However his proposed scheme, although an interesting compromise between ambiguity and geometric competence, neither reduces the ambiguity to the point of corresponding to human perception, nor fully enforces the applicable geometric constraints. An obvious suggestion then is to look for a way to select the interpretation which conforms or most nearly conforms to the Huffman-Clowes restrictions while retaining the more general geometric powers of sidedness reasoning for use as and when necessary.

This paper outlines an approach which uses models of familiar objects and fragments of objects to implement, and hopefully to improve on, this suggestion. The program will be based on the sidedness reasoner from Ellsid, which is essentially rule-based, but will be augmented by the models whose role is to guide the interpretation - in effect resolving the ambiguity found by Ellsid alone. It is argued that this offers a method of capturing the good aspects of past programs (including the Huffman-Clowes scheme) particularly their abilities to choose the same interpretations as people, while overcoming their inadequate grasp of geometric constraints and inability to cope with accidentals when this is necessary. In addition it offers a way of modelling the effect of common or familiar objects or configurations on the perception of drawings.

Model-based vision programs

Most, perhaps all, programs can be seen as model-based in some sense despite the fact that their general "feel" may be that of a bottom-up general purpose method. For instance Woodham's (1977) program for getting shape from shading by a local computation uses models of fragments of surface shape. In the domain of line-drawing interpretation, the succession of programs can be seen as having been based on successively smaller models. Roberts (1965) used complete, simple, convex polyhedra such as bricks and wedges as models. Line-labelling schemes are based on models of possible vertices and their appearances. Mackworth's Poly and the author's Ellsid both use planes as their basic element or model: they reason about how these may be fitted together to make up scenes.

Here it is useful to distinguish three different aspects of the use of models.

1. They can largely determine the way in which the scene (i.e. the interpretation) is described. This is most obviously true when models are used to achieve recognition of known objects - the interpretation may then consist almost entirely of the names of those objects. Likewise when models carry information which could not otherwise be deduced from the picture - such as lengths in Falk's (1972) program and hidden surfaces in Roberts' program - they have a large effect on the content of the scene description. Apart from this effect on its content, models may affect its structure by determining the elements of which it is made up: Roberts' program sees an L-beam as two bricks welded together whereas Poly sees it as planes meeting along edges with no sub-division into convex blocks.
2. Models can be used as hypotheses - sets of conclusions about the scene that are jumped to on the basis of slight evidence, though some checking may follow. Roberts' program has this flavour; line-labelling does not since it considers all possibilities and allows all consistent interpretations that survive all the checks it knows how to make. As we shall see, it is this aspect of models that is needed in the present application since they are to be the basis for going beyond the geometric constraints.
3. Models are often the basis for organizing the way knowledge is built into the program - primarily constructs to help the programmer organize the code, nuclei for structuring the program. It is in this sense that all the programs mentioned are model-based - they are all organized around some basic scene concepts. Loosely speaking, frames (Minsky 1975) are models in this sense since a frame brings together procedures as well as declarative information, and the idea is to organize the program round frames whether or not

it has a hypothesise-and-test flavour and however its scene descriptions are organized. In vision this is taken furthest by Freuder (1976) who organizes all knowledge around models but these appear in what are effectively four different networks of declarative information and have at least two sets of procedures associated with them (for use when activated in top-down and bottom-up modes respectively).

A program design using models to select interpretations

We are now on a better position to specify clearly what role the models are to play in the proposed program design. We do not want them for defining the scene description language - that is still to be done primarily by the underlying sidedness reasoning program using line-labels and its representation of planes and their relationships; neither are the models to be central units of the program organization. They are instead required to provide likely hypotheses about parts of the scene.

The basic sidedness reasoner is retained firstly to ensure that the final interpretation is geometrically consistent - and so it is used to check that the hypothesised fragments of scene description are mutually compatible. Each hypothesis must therefore be expressed in terms that the reasoner can deal with directly - as sidedness assertions about planes. Note that in terms of the underlying plane-based approach the models do not correspond to natural fragments of scene objects: for instance the all-convex labelling of a Y-junction, which in the Huffman-Clowes scheme is a complete model of a vertex, tells a plane-based system something about one corner of each of three surfaces and the way they meet each other there. In this respect these models are more like the M.I.F.s and M.U.F.s developed by Frank Birch (1978) in a letter recognition system than they are like Roberts' models. M.U.F.s (minimal unambiguous fragments) are combinations of strokes that have no meaning by themselves as letters but are valuable to a program as a combination that can belong to only one letter and can hence initiate some special processing. M.I.F.s (minimal impossible fragments) are stroke combinations that cannot be part of any single letter and hence signal that some stroke junctions must be undone. Both are models selected not for their significance in the resulting interpretation but for their usefulness to the process constructing the interpretation.

The second function of the sidedness reasoner is to fill in the gaps in the interpretation when there are parts of the picture not covered by hypotheses of familiar configurations - e.g. at accidentals. How many such "gaps" occur - that is, how much of the interpretation will not be covered by the models - depends partly on how extensive the set of models is, and partly on whether the control strategy is to search for the interpretation that has the maximum proportion supplied by models, or to grow an interpretation outwards from an initial model match, using other models if possible but only backtracking if forced to by a geometric inconsistency. The latter strategy will not always give the "best" interpretation and in general will depend on the order in which parts of the picture are tackled. Probably both should be explored to see which can be made to fit human performance best.

A second major design decision is to trigger the models by matching cues in the picture domain. It is possible to have a system where scene descriptions are generated by some means and then models of 3-D configurations are matched to them. This is sufficient for recognition systems, where the purpose of the models is to identify known objects, and it could be used here by preferring interpretations containing familiar 3-D configurations. However it seems unlikely to provide a good model of human preferences for several reasons. Firstly, such a system would behave like a paranoid: it would have strong ideas about what the interpretation "should" be and very slight evidence would be enough to "confirm" this - it would take no account of what those appearances probably or usually indicate. What we really want is a system that assigns appearances a plausible interpretation and does not invoke unlikely links between appearances and interpretations unless it has to. This is achieved by having a stored set of picture configurations called "keys" or "cues" each of which triggers a particular fragment of 3-D interpretation called a model. It is these pairs of keys and models that are stored, and have been loosely referred to up to now simply as "models".

This decision fits well with previous programs, and with the aim of trying to recapture the insights contained in past work. Line-labelling is based on an insight first exploited by Guzman (1969) that picture junctions are good evidence about the scene (see Hochberg 1968 for evidence of the psychological reality of this): what we want to capture here is the idea that Y-junctions, say, normally have one of the three interpretations allowed by the Huffman-Clowes scheme although a lot more are possible. Similarly in Roberts' program models have keys associated with them that are good candidates for incorporation in the present program, as are the "line features" used by Grape (1973).

Principles for choosing model-key pairs for incorporation

Depending on the motives for constructing a version of the proposed program, various principles might be used in selecting the model-key pairs to be used. Interesting experiments could be made to see how well the ideas in previous programs will work when augmented by sidedness reasoning - for instance by using as models just the Huffman-Clowes set of junction labellings one could find out if it could now cope with multihedral vertices and accidentals while avoiding geometrically impossible interpretations and still generally producing only those interpretations which people see.

Another idea would be a learning program which had some method of analysing each picture-plus-dictated-interpretation in a training sequence and compiling a set of picture fragments plus interpretations to use as models, selected perhaps on the basis of frequency of occurrence. Alternatively a set could be compiled by hand, the aim being to model the interpretations people select over as large a set of pictures as possible.

The above program designs might possibly achieve a good approximation to human perceptual behaviour (within the limits of the power of line-labels to express 3-D interpretations) but they could never offer a theoretical explanation of why they worked. What than should the principles for selecting models be? A possible answer comes by extending Huffman's General Viewpoint idea (Huffman 1971 p.298): we want picture-to-scene hypotheses that are probable. To explore this we need to develop the picture/scene distinction emphasized by Clowes (1971).

The keys are defined in the picture domain - e.g. a Y-junction is defined by picture angles etc. independent of the labelling (interpretation) that may later be assigned to it. The models are defined in the scene domain - that is they specify aspects of the 3-D scene in a way that in principle is independent of the picture (this is completely true of Roberts' program, only partly true when the interpretation is specified by line-labels). In order to identify probable key-model pairs we must consider the relationship between parts of the scene and the appearances they generate - I shall call these pairings "mapping events". A good example of a mapping event is a T-junction generated by an edge being occluded. Not all T-junctions signal occlusion and not all occluded edges generate T-junctions. When trying to interpret a T-junction the question is: was occlusion the mapping event that generated it? If that is the hypothesis you adopt, it dictates certain features of the putative scene - a relative depth relationship for instance; in general several somewhat unconnected scene relationships are specified by the hypothesis of a given mapping event. Occlusion is not a scene property - it is not part of a scene but a consequence of the association of a scene and a viewpoint, pairing a particular appearance (i.e. a picture) with the scene.

In order to pursue the idea of choosing key-model pairs that represent probable hypotheses, then, models should not be chosen by selecting convenient scene fragments nor keys by selecting convenient picture fragments. Rather we want to identify keys that have a good chance of leading to a correct piece of interpretation. It is mapping events like occlusion and accidentals that are more or less probable, and so it is these probabilities that should be taken into account in formulating hypotheses and in designing the model set that determines the hypothesis-making of a program. Key-and-model pairs correspond to mapping events and should therefore be selected for maximum probability of the correspondence. Some mapping events have a relatively high probability and are worth being stored explicitly as models, while others do not have a high enough relative frequency to warrant this. Ideally we would like to identify keys for which a subset of their possible interpretations account for the large majority of their preferred interpretations in practice. The non-accidental interpretations of a junction are an example of this. The selection should also be influenced by the effect of other constraints operating on a hypothesised interpretation - one can afford to consider fairly unlikely interpretations if they are (nearly) always ruled out by other constraints when they are incorrect (not preferred). A considerable further amount of theoretical work needs to be done on the behaviour of such systems, but practical experience with various sets of models in the proposed system may suggest important leads in this.

Conclusion

The proposed program will have a competent geometric facility as its basis (the sidedness reasoner) to check the overall interpretation produced and to act as a medium for combining the contributions of other parts. This will be supplemented by a system of stored models of a range of sizes. These models are not included to provide the program's basic ability at interpretation but to generate good hypotheses for the reasoner to work on. They can be seen as partial results stored ready-made because they are frequently encountered, and as such they may save computation. However that is not their primary function. Like M.I.F.s and M.U.F.s, they are items useful to the program for controlling the computation rather than for making up large portions of the output. They are hypotheses in the sense intended by R.L. Gregory (e.g. 1974) when he characterised vision as a process of forming and checking hypotheses: they are used to resolve the ambiguity inherent in the picture. They cause the program to jump to a conclusion that goes beyond the evidence in the sense that, although it will check that no constraint refutes the hypothesis, the interpretation is partly determined by ignoring the possible alternatives.

Acknowledgements. The ideas put forward in this paper mostly originated in some form with my colleagues. Max Clowes showed me that most vision programs can be seen as based on models of some sort. Aaron Sloman has persistently argued for the simultaneous use of models of different sizes: an idea analogous to that put forward for natural language by Becker in "The phrasal lexicon". But most immediately important was talking to Dave Owen while he struggled to find principles for applying this idea in the Popeye program: it was he who made me realise that models may not form natural subdivisions in the scene domain.

The research of which this forms a part was supported by the Science Research Council, grant number GR/A79703.

References

Becker J.D. 1975 "The phrasal lexicon" in TINLAP conf. proc. ed. Schank R. and Nash-Webber B.L. pp.70-73 (Cambridge, Mass.: Bolt, Berenek, and Newman)

Birch F. 1978 "A (self-adapting) network for recognition of visual structures" Proc. A.I.S.B./G.I. conference, Hamburg.

Clowes M.B. 1971 "On seeing things" Artificial Intelligence vol.2 no.19 pp.79-116

Draper S.W. 1978 "Competence at interpreting line-drawings: a preliminary report on Ellsid" Report, Cognitive Studies Programme, Sussex University.

Falk G. 1972 "Interpretation of imperfect line data as a three dimensional scene." Artificial Intelligence vol.3 no.2 pp.101-144

Freuder E.C. 1976 "A computer system for visual recognition using active knowledge" AI-TR-345 (Cambridge, Mass.: M.I.T.)

Grape G.R. 1973 "Model based (intermediate-level) computer vision" Stanford AI memo AIM-201 computer science department, Stanford University

Gregory R.L. 1974 "Perceptions as hypotheses" in Brown S.C. (ed.) Philosophy of psychology (London: MacMillan) pp.195-210

Guzman A. 1969 "Decomposition of a scene into three-dimensional bodies" in Grasselli A. (ed.) 1969 Automatic interpretation and classification of images pp.243-276

Hochberg J.E. 1968 "In the mind's eye" in Haber R.N. (ed.) Contemporary theory and research in visual perception pp.309-331 (London: Holt, Rinehart, and Winston)

Huffman D.A. 1971 "Impossible Objects as Nonsense Sentences" in Machine Intelligence 6 pp.295-323 ed. Meltzer B. & Michie D. (Edinburgh: Edinburgh University Press).

Kanade T. 1978 "A theory of origami world" Technical report, dept. of computer science, Carnegie-Mellon university. CMU-CS-78-144

Mackworth A.K. 1973 "Interpreting pictures of polyhedral scenes" Artificial Intelligence vol.4 no.2 pp.121-137

Minsky M. 1975 "A framework for representing knowledge" in Winston P.H. (ed.) 1975 The psychology of computer vision (New York:McGraw-Hill) pp.211-277

Roberts L.G. 1965 "Machine perception of 3-D solids" in Tippett et al. (eds.) Optical and electro-optical information processing pp.159-197 (Cambridge, Mass.: M.I.T. Press)

Waltz D.L. 1972 "Generating semantic descriptions from drawings of scenes with shadows" MAC AI-TR-271 (Cambridge, Mass.: M.I.T.)

Woodham R.J. 1977 "A cooperative algorithm for determining surface orientation from a single view" Proc. I.J.C.A.I.5 vol.2 pp.635-641