COOPERATIVE SYSTEMS FOR PERCEPTUAL TASKS IN A REMOTE SENSING ENVIRONMENT

Alan K. Mackworth

Fellow, Canadian Institute for Advanced Research
Department of Computer Science
University of British Columbia
Vancouver, B.C. V6T 1W5

Abstract

To design and implement knowledge-based systems for perceptual tasks, such as interpreting remotely-sensed data, we must first evaluate the appropriateness of current expert system methodology for these tasks. That evaluation leads to four conclusions which form the basis for the theoretical and practical work described in this paper. The first conclusion is that we should build 'cooperative systems' that advise and cooperate with a human interpreter rather than 'expert systems' that replace her. The second conclusion is that cooperative systems should place the user and the system in symmetrical roles where each can query the other for facts, rules, explanations and interpretations. The third conclusion is that most current expert system technology is *ad hoc*. Formal methods based on logic lead to more powerful, and better understood systems that are just as efficient when implemented using modern Prolog technology. The fourth conclusion is that, although the first three conclusions can be, arguably, accepted for high-level rule-based symbol-manipulation tasks, there are difficulties in accepting them for perceptual tasks that rely on visual expertise. In the rest of the paper work on overcoming those difficulties in the remote sensing environment is described. In particular, the issues of representing and reasoning about image formation, map-based constraints, shape descriptions and the semantics of depiction are discussed with references to theories and prototype systems that address them.

Keywords: Cooperative systems, expert systems, map interpretation, remote sensing, perceptual tasks, shape description, depiction.

1. Introduction

The aim of this paper is to present a research program underpinning the design and implementation of cooperative systems for perceptual tasks in a remote sensing environment. The need for the program is motivated by an analysis of current expert system technology and the requirements of perceptual tasks.

2. Expert Systems

Expert system technology has extended the range of problems amenable to computer-based solutions. The classic view of an expert system is that it is a computer program that satisfies the following requirements (Walker *et al.,* 1987). It must solve, or help to solve, an important problem that would otherwise require human expertise and judgement. It must integrate new knowledge incrementally into the knowledge base. It must help the designer and user elicit, organize, display and transfer knowledge. It must provide explanations of its advice. It must reason with inexact and exact knowledge. Finally, it must support a readable and natural user interface. To satisfy these requirements an expert system must have a knowledge base of facts and rules, and a rule engine for deriving new facts and establishing goals. In addition it may have an explanation generator, methods for acquiring and encoding new knowledge and a dialogue handler for the user interface.

The standard task classification for expert systems (Stefik *et al.,*1982) breaks the applications into the following six generic task domains: interpretation, diagnosis, monitoring, prediction, planning and design. Problems in these domains have in common the characteristic that their space of possible solutions is very large, ruling out a generate-and-test exhaustive enumeration algorithm. In addition the tasks may require tentative reasoning with incomplete knowledge on time-varying, noisy or incomplete data.

On the face of it then, current expert system technology, as embodied in rule-based shells, is ideal for interpretation tasks in a remote sensing environment such as, say, updating a forest cover map based on satellite imagery, a digital elevation model and an earlier forest cover map. But, in fact, that is far from the case; we shall examine why this is so and what can be done about it.

3. Cooperative Systems

There are essentially two reasons why current expert system technology is inappropriate for most perceptual tasks. First, it doesn't work very well. Second, it takes us in the wrong direction. It doesn't work for a variety of reasons, but basically because it fails to build internal models of the process it is trying to understand.

Attempts to build diagnosis systems that reason about the structure and function of the device to be diagnosed are leading to better performance. This is known as 'diagnosis from first principles' or 'model-based reasoning'. We must do the same for interpretation tasks.

When I say it takes us in the wrong direction, I mean simply that trying to build a program to replace a human expert represents an attempt to de-skill the task. It is not likely to lead to strong support from the expert; moreover, if the program cannot interact in terms of models the expert understands the prospect of effective knowledge transfer is minimal.

This rationale lies behind arguments for designing and building prototype 'cooperative systems' that advise and cooperate with an expert, or a novice, interpreter. Such systems are 'cooperative' in another sense as well. We envisage a cooperative system having a variety of knowledge sources, *including the user*, and allowing them to cooperate to arrive at a mutually consistent interpretation.

For example, given that standard maximum likelihood methods can produce partially correct classification results based on spectral signatures (subject to several restrictive assumptions), the user could sketch a map on the image, allowing a sketch map interpretation program to interpret the map, supplying spatial constraints and context sensitivity to a spectral segmentation knowledge source. Such a cooperative system has been realized (Glicksman, 1983; Havens and Mackworth, 1983).

For such systems to succeed, the user, who is an active participant in a cooperative system, must come to trust the other components. This will only occur if the other components are seen to be *transparent* and *reliable*. Perhaps the only way to ensure this is to place the user and each component in symmetrical roles where each can query the other for facts, rules, explanations, interpretations and justifications. This approach to interfaces for knowledge-based systems has been called the QUARFE interface since it is based on a protocol that allows Questions, Answers, Rules, Facts and Explanations (van Emden, 1988).

Much current expert system technology is *ad hoc*. The programs are large, complex, opaque, and unreliable. When an answer is computed its relation to the input is unclear and when new rules are added they interact with old rules in unexpected ways. These problems lead to major difficulties in scaling up from small projects to large ones. By implementing the rule interpreter in Prolog as an extension of the standard Prolog interpreter (Sterling and Shapiro, 1986) one can add facilities that implement the QUARFE interface. This also allows one to know that a conclusion drawn by a component is a logical consequence of its facts and rules. This enables the user to trust the component because it is transparent and reliable. Moreover, one can implement, in the rule interpreter, schemes for representing inexact knowledge based on a formal Bayesian theory of probability without paying the price of losing the clear semantics of first order logic (Poole, 1989).

4. Remote Sensing Tasks Requirements

Although the conclusions of the previous section regarding the need for, and structure of, cooperative systems may be, arguably, accepted for high-level symbol manipulation tasks there are major hurdles on the path of implementing them for perceptual tasks that rely on visual expertise. In this section we mention some of those difficulties and describe some work on overcoming them.

A cooperative system whether it be for diagnosis or interpretation must represent, and reason about, the underlying physical reality it is dealing with by constructing an adequate model of that reality. For perceptual interpretation tasks the image formation process must be thoroughly modelled. We cannot make simple assumptions such as assuming that image irradiance is a function solely of a scalar surface albedo which uniquely characterizes the ground cover. Models of the optics of image formation including distributed illumination sources, scattering and haze, surface elevation, slope and aspect, non-Lambertian surface reflectance and so forth as, for example, in (Woodham *et al.*, 1985) and

(Woodham and Gray, 1987) are essential.

These models capture the optical constraints of the task, and they are necessary but they are not sufficient. There are often non-optical, physical and perhaps even cultural, constraints that must be modelled. The user and the various components of the cooperative system must be able to communicate questions, answers, rules, facts and explanations about such constraints easily. Although, a component can model these constraints *internally* using sentences in predicate calculus, at the user interface they must be represented visually using the graphical formatting conventions that have developed into the language of maps.

In our Mapsee project we have designed, implemented and tested a series of computer programs, Mapsee-1, Mapsee-2 and Mapsee-3, for interpreting sketch maps of geographic regions. Mapsee-2 and Mapsee-3 use schemas to represent and reason with a variety of constraints (e.g. 'roads must be on land', 'a shoreline bounds a land region on one side and a water region on the other', . . .) to achieve a consistent interpretation of the map. See (Mulder *et al.*, 1988) for an overview of the Mapsee project. As mentioned in Section 3, Mapsee has been used to provide a visual interface to an interpreter, allowing him or her to sketch features over aerial images of small towns (Glicksman, 1983). The Mapsee interpretation of the sketch is then used to provide tight contextual constraints to a traditional maximum likelihood classifier. This illustrates the cooperative principle: the user and the system components each contribute their expertise to arrive at a shared goal.

In general then, a cooperative system for remote sensing tasks must accept images in various forms as 'facts' from the user. In order to be useful these images must be referred to a canonical coordinate system and interpreted into a common framework. But we should not require the user to provide a map registered to an image or to carry out manual registration through the use of ground control points. Sensor-based raster imagery must be automatically registered to map-based vector data. Both the raster imagery and the vector imagery may be provided at many different scales. Good descriptions and multiscale matching techniques are necessary for registration. Recently we have developed techniques that smooth contours in a path-based coordinate system. These techniques have certain properties that are essential for matching purposes (Mackworth and Mokhtarian, 1988). We have also developed a system that exploits this representation to achieve automated registration of Landsat MSS data to a map database (Mokhtarian and Mackworth, 1986).

The user interacts with a perceptual cooperative system visually – by drawing and reading sketch maps, for example, rather than typing sentences. On the other hand, we concluded in Section 3 that such a system should use a formal logic as its internal representation language. (Or, at least, it can be characterized as so doing – it may not be implemented that way.) This leaves a chasm to be bridged: we need a logical theory of the semantics of maps. We describe such a theory for diagrams in general in (Reiter and Mackworth, 1989). This logical framework requires the implementer to write a set of sentences in first order predicate calculus to describe the image, another set to describe the scene and a third set to describe the image-scene depiction mapping. These sentences can include general facts about a class of images (the set of all maps, say) – what image objects can appear in them, how they relate and so on – and also facts about a particular image. An interpretation of a diagram is defined to be a logical model of the complete set of sentences. An implementation of this theory now exists in prototype form for a restricted class of maps.

5. Conclusion

In summary, knowledge-based systems for perceptual tasks should be designed as cooperative systems not expert systems. Cooperative systems should be seen as including the user. The components of such a system interact in symmetrical ways using the QUARFE protocol. Internally, the components should be specified and, perhaps, implemented using first order predicate calculus.

An analysis of the obstacles preventing the implementation of such systems for perceptual tasks in a remote sensing environment was presented. This analysis established four research goals. The first is to build adequate models of image formation. The second is to represent and reason about map-based constraints. The third is to derive multi-scale shape descriptions and matching algorithms. The fourth is to establish a theory of depiction that applies to maps and other diagrams. The resulting theories and prototype systems demonstrate that the goal of building cooperative systems for perceptual tasks is achievable.

6. Acknowledgements

The financial support of the Natural Sciences and Engineering Research Council of Canada and the Canadian Institute for Advanced Research is gratefully acknowledged. I am also grateful to the faculty, staff and students of the Laboratory for Computational Vision for ongoing dialogue and support.

References

1. Glicksman, J., "Using multiple information sources in a computational vision system", *Proc. IJCAI-83*, Karlsruhe, W. Germany, pp.1078–1080, 1983.
2. Mackworth, A.K. and Mokhtarian, F. "The renormalized curvature scale space and the evolution properties of planar curves", *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Ann Arbor, MI, pp.318-326, 1988.
3. Mulder, J.A., Mackworth, A.K., and Havens W.S., "Knowledge structuring and constraint satisfaction: the Mapsee approach", UBC Dept. of Computer Science, TR87–21, 1987.
4. Poole, D. "Computer Science 532 course notes", Dept. of Computer Science, University of British Columbia, 1989.
5. Reiter, R. and Mackworth, A.K. "A logical framework for depiction and image interpretation", UBC Department of Computer Science, TR88–17, to appear in *Artificial Intelligence*, 1989.
6. Stefik, M. *et al.*, "The organization of expert systems, a tutorial" , *Artificial Intelligence* 18, pp.135–173, 1982.
7. Sterling, L. and Shapiro, E. *The Art of Prolog: Advanced Programming Techniques*, MIT Press, Cambridge, MA, 1986.
8. van Emden, M.H., "Project 4.5", University of Victoria, Victoria, B.C., 1988.
9. Walker, A., McCord, M., Sowa, J.F. and Wilson, W.G. *Knowledge Systems and Prolog*, Addison-Wesley, Reading, MA, 1987.
10. Woodham, R.J., Catanzariti, E. and Mackworth, A.K., "Analysis by Synthesis in Computational Vision with Application to Remote Sensing", *Computational Intelligence*, 1(2), pp.71–79, 1985.
11. Woodham, R.J. and Gray, M.H., "Analytic method for radiometric correction of satellite multispectral scanner data", *Proc. IEEE-GE*, vol.25, pp.258–271, 1987.