A Frame-based System for Modelling and Executing Visual Tasks

Peter W Woods, David Pycock, and Christopher J Taylor

Wolfson Image Analysis Unit
Department of Medical Biophysics
University of Manchester, Oxford Road
MANCHESTER M13 9PT

This paper describes a framework for model representation and control intended as the basis of a computer vision system capable of undertaking arbitrary but well defined image interpretation tasks. An inter-related structure of models represents both the visual task and image content. The user defines this structure but not the internal details of all models, so non vision experts can program the system. The approach is illustrated by considering two examples, one in interpreting cell images, the other in industrial inspection.

INTRODUCTION

This work forms part of a project called 'Techniques for User-Programmable Image Processing' (TUPIP), which is directed towards producing a computer vision system which can accept a description of a visual task from a user who is not an image processing expert and generate a 'solution', ie be capable of performing the task. The chief requirements of such a system are that there is sufficient flexibility for a user to adequately describe arbitrary image contents and analysis goals, and that there is a reasonably efficient but task independent control strategy that can execute the task robustly. This paper describes a model-based framework for knowledge representation and control that can fulfill these demands. We exploit the fact that the specific nature of a well defined visual task enables a detailed model to be constructed. The types of visual task considered involve static images of repetitively occurring scenes.

Whereas model-based approaches to computer vision are widespread [1], most do not use prior knowledge to direct the image processing throughout their execution, but rely on a data-base of features extracted at the start. This is acceptable if the features can be chosen in advance to be appropriate to the type of image and the visual task. In a general purpose system, it becomes unfeasible to measure all the potentially useful image properties beforehand, so there must be a mechanism for gathering evidence on demand. We have chosen a framework for representation based on frames in the sense of Minsky [2] which allows methods for acquiring data to be associated with the data.

This work is supported by an SERC grant and is carried out as part of Alvey project MMI-093 : 'Techniques for User-Programmable Image Processing'.

Frame-based representations have become increasingly popular for AI in general and computer vision in particular [1, 3, 4] . The VISIONS system described by Hanson and Riseman [5, 6] is an example of a frame-based system which interprets outdoor scenes by region labelling. They recognise the need for goal directed control of low level processes and use properties of region types which are acquired from examples to select dynamically the operation which will give the best resegmentation of a doubtful region. This mechanism is not however general enough to be useful in other types of visual task.

Tsotsos has described a highly structured framework for modelling image sequences [7] which has been applied to the interpretation of coronary radiographs [8]. The model structure and control strategy are generic, but the application relies on the programmer providing detailed internal definitions of frames including application specific instantiation methods.

We describe a framework for knowledge representation, similar in some ways to these, which allows fairly straightforward representation of prior knowledge about the task. The user defines an inter-related set of sub-models which are frames, arranged in a multi-levelled structure which defines the task and also models the image contents. In this paper, the entire structure will be referred to as the 'world model', and sub-models will be referred to as model elements. The structural relationships are defined by part-of links augmented by descriptions of inter-relationships such as relative position, whereas individual model elements are defined as specialisations of generic system-defined model elements called prototypes. In a particular task, the structure of the world model is used in conjunction with a task independent control strategy to guide its instantiation and this process is equivalent to achieving the goals of the task.

The user of our system is expected to have sufficient insight into the visual task to specify the structure of a world model. It cannot however be assumed that the user is able to specify the detailed properties of model elements nor describe how these are to become instantiated. The problem thus arises as to how methods of instantiation can become associated with user-defined model elements. We describe below how this problem can be overcome by specialisation. Many model elements will contain parameters associated with the description of such things as shape, grey-level properties,

AVC 1988 doi:10.5244/C.2.8

and spatial relationships which are implicit within prototypes and initially undefined. We also believe that the only sensible way for the user to describe properties or variants of some model elements is by the use of examples. A training session is therefore considered essential in creating a useful world model.

A mechanism to control instantiation of the world model is crucial because even where the imaged scenes are highly constrained the number of possible interpretations of an image will be far too large to be exhaustively searched. The two most important issues are how to guide the pattern of instantiation so as to reach an optimal interpretation in an efficient way, and how to reason with uncertain information. The latter issue arises since it must be assumed that image features are subject to distortion and loss, and there may be a significant degree of variability in the appearance or form of the imaged scenes from one instance to the next.

We do not yet have a fully integrated implementation of our system although individual parts are currently being tested. However, we describe in detail how the framework we have outlined can be applied to two example problems, one involving medical image interpretation, namely chromosome analysis, and the other from the realm of industrial inspection.

MODEL REPRESENTATION

The World Model

We propose a world model which has two roles. The first is to represent items expected to be encountered in images, to allow their detection and recognition. The second role is to model the task. We assert that the world model should be structured so that these roles are combined and, in effect, the instantiation of the complete model is equivalent to achieving the goals of the task. In this sense, the proposed model differs from other frame-based approaches such as VISIONS which focus on the structural description of image content. In our system, goals are represented by model elements which the user indicates explicitly to be the goal frames. These might represent a fault report, a list of derived measurements, or a synthesised image as in the examples below. The elements of the goal frames are related either directly or via one or more levels of abstraction to model elements which do represent image content.

A multi-levelled structure for the world model is of importance for several reasons. It allows the potential computational complexity of the visual task to be limited by constraining the number of interpretations which need to be considered at any stage. It also makes the specification of a complex visual task more manageable and allows results to be directly associated with what the user has defined. The principal type of structural link used to build a world model from model elements is the 'part-of' link which allows model elements (frames) to contain other frames. The part-of link has a dual role in representing physical containment in the imaged scenes and also representing ownership of abstract properties. There must also be the ability to represent constraints on relationships of various types between model elements. These relationships are expressed by other frames with links to two model elements and are considered in detail below. The only other type of link required is a reference link whereby frames can contain references to slots of other frames, allowing data to be shared. This is necessary because methods of frames cannot refer directly to other frames if they are to be independent of the world model structure. (An exception to this is when methods involve model elements which are always present, such as instantiation methods which use both grey-level and shape sub-components.) Reference links allow specific values which correspond to properties of one model element to be used by methods belonging to another model element.

The lowest level of model elements in the world model can represent fairly complex image structures - the model does not extend explicitly below what the user would normally consider as objects, eg as far as edge segments or other low-level image primitives. Shape descriptors capable of representing objects are provided by specialisation. These can contain more than one representation of the object and are capable of transforming between representations [9]. Neither the shape descriptor nor therefore the corresponding model element can represent holes or subregions within a shape and so these have to be modelled by part-of links with the holes or subregions represented by their own model elements.

Specialisation

The system provides a hierarchy of generic prototypes. The user is able to select one of these as a basis for a more specific definition, ie the user can generate subclasses for specific cases which can inherit appropriate representational structures and associated methods. This means that the user does not need to specify the internal forms of shape models and other generic properties. Obviously the range of tasks that the system is capable of performing depends on how broad a set of prototypes are available, and they must be documented in a way which will allow the user to select them intelligently. However the library of prototypes can be easily extended without the structure of the system needing to be changed.

Any model element representing an object in an image will be a specialisation of some 'object-frame' prototype. All such model elements will inherit a shape descriptor together with positional and size descriptors, an

area grey-level descriptor and a boundary grey-level descriptor. (The latter is a model of the grey-level profile across the object boundary). All model elements will also inherit a set of properties concerned with control. These are a confidence factor, representing the quality of instantiation or closeness to expectations, and two subsidiary quantities, 'importance' and 'utility'. Importance is a measure of how necessary the model element is for the achievement of the goals. Utility is a measure of how useful a model element is in helping to instantiate the model elements to which it is related. This is a function of the degree to which it constrains the properties of related elements, the expected quality of its instantiation, and also the computational cost of its instantiation though this is not taken into account at present.

Specialisation occurs when the world model is first created, before the system executes. The system does not generally explore the specialisation hierarchy except where there are different subclasses of some common prototype explicit in the world model. Specialisation consists mostly of constraining the properties of a superclass, but new properties can be added via the part-of relationship. For example, a 'symmetric-ribbon' prototype is a specialisation of the object-frame prototype whose shape model has certain prior constraints.

Relationships

The types of relationship that can exist between model elements which represent image content are spatial, including both topological and geometric relationships, and conditional, including logical and cardinal properties. Constraints on spatial relationships have obvious uses both in limiting the area of an image that needs to be processed to find evidence for a particular object, and in reducing the number of interpretations of generated evidence that need to be considered. Thus they have a crucial role in controlling instantiation. Relationships are defined only between a parent model element and its sub-components or between sibling model elements.

Geometrical relationships are represented independently of the nature of the objects involved so that common methods can be used to combine different constraints on object positions. In the TUPIP project it is assumed that 3-D scenes can be interpreted in terms of 2-D models and so the geometrical relationships are confined to a plane, and are expressed in terms of relative position in polar coordinates, and relative orientation. Topological relationships are limited to touching, overlapping, containment, and their inverses.

The user must indicate that spatial relationships do exist between specific model elements, but need not specify them completely since this can be done by the system itself during training. The system does not attempt to discover for itself all the possible constraints that might operate between model elements and assumes there are none if the user does not specify otherwise.

Whereas spatial relationships are defined with respect to the positional attributes of model elements, conditional relationships are qualifications of part-of relationships. Conditional relationships are required to capture information on the cardinality of model elements and whether the element must be or might be present in a scene. They are also used to express conditions on the properties of model elements imposed by others. Examples are: "There should be two instantiations of chromosome #1", and "There is either one X-chromosome and one Y-chromosome in the image, or two X-chromosomes". Relationships of this type are both provided by the user and internally represented in a declarative form.

TRAINING

The world model can be considered to be 'compiled' from the user's definition into a static structure. Before this can be used to guide execution of the task, implicit parameters within model elements must be set up and this is achieved by a training phase. Training operates in a similar manner to that described by Woods *et al.* [10] using a representative set of example images. The training phase consists of a series of step by step attempts at instantiation with results being displayed to the user in an appropriate way. The first training pass involves a considerable amount of explicit questioning of the user by the system, for instance to prompt the user to indicate examples of particular model elements in an image. This will allow an initial set of parameters to be derived which can be used for future attempts at instantiation and refined subsequently. The user is prompted to confirm or modify results at each step. For example, after instantiating the shape of an object, the boundary might be displayed overlayed on the original image. The user can either accept this or move part of the boundary with a mouse to indicate an acceptable result. Similarly precursory possible sites of objects provided by cue generators can be indicated as true or false. In this manner it is intended that 'hidden' parameters of grey-level properties, geometrical constraints, and shape descriptors can become fully defined for all of the model elements for which they are relevant. The parameters represent not only average values but also allowed ranges and degrees of variability.

During training, statistics on confidence factors are gathered both for successful and incorrect instantiations and from these a value can be obtained which is used to normalise the factors obtained during execution. Training also serves to define values for importance and utility. These can each take on one of a

small set of possible values and are initialised to default values when the world model is created. During training they relax to appropriate values which remain constant thereafter. For example, where inherited model sub-components are irrelevant to a particular task their importance value will become set to the minimum value so that no attempt will normally be made to instantiate them. Conversely elements which are essential to fulfilling the goals of a task will have the highest importance. Utility values will be based on the reliability of instantiation as measured during training.

A training session can fulfill a complementary role to that of collecting parametric data, in delineating actions to be associated with particular points of instantiation or conflict between interpretations. For example when a candidate object is too large and has an abnormal shape, an attempt should be made to interpret it as two or more candidates. Even assuming a generic 'splitting' method exists to seek evidence for a disjunction between two points or regions, this must be selected from other such methods. This will rely on the user manually indicating and splitting such aggregates during training.

The system becomes more reliable as the training session progresses and can be terminated by the user when appropriate. However the system will normally be able to revert automatically to training mode if it detects a situation where it cannot produce an acceptable interpretation.

CONTROL

We have already introduced the idea that the goal of a visual task is represented as one or more identified frames in the world model. The instantiation of these frames provides the fulfillment of the goal and will in general depend on the instantiation of the rest of the model. The matching of derived data to model elements is not a simple graph matching process but must take into account the variability and uncertainty in both model and image data.

Initially, control involves backward chaining from the goal frame(s) to successive part-of elements in the model structure until an element is encountered which can be directly instantiated. Instantiation is then attempted, which typically involves cues being generated from the image within an area delineated by the known spatial constraints, and each cue in turn being assessed as genuinely arising from the expected object. This can lead to one or more candidate instantiations of the relevant model element. Cue generation methods [11, 12] are built into prototypes and are controlled by the shape and grey-level parameters of the model element. The cues are designed to correspond directly to one form of representation used by the shape descriptor [9]. Verification consists of using the cue to gather more detailed and specific data which can be matched against the model to produce an interpretation and an associated confidence factor. The confidence factor depends on how well the parts of the model element were themselves instantiated and also on the importance factor associated with each part. Both utility and importance aid efficiency by helping to select the best route for instantiation, but are not essential for the system to discover an interpretation eventually.

This pattern of cue generation and verification is one level of a 'hypothesise and test' control strategy. Once some model element is instantiated, it can become possible or easier to instantiate related model elements and the control strategy changes to a data-driven mode which is, however, constrained by the relationships in the world model. There is not in general a modelling of the whole of the image and only features that are intended to match specific model elements are sought.

Once the sub-components of a higher level model element have been instantiated, the model element itself can become instantiated. We will say that a model is instantiated when an adequate match is obtained between an adequate number of the components of the model and candidate data. The hypothesise and test strategy is modified when verification fails or when a higher level inconsistency arises, ie when evidence may accumulate for believing that certain locally acceptable interpretations are inconsistent. In such circumstances a search for confirmatory evidence is undertaken. The exact operations to be performed when such an inconsistency is discovered must be limited in range and there must be criteria for choice which will come in part from the training session as outlined above. The resulting new or confirmed interpretation is then used as before in a data-driven phase.

IMPLEMENTATION

We have considered the use of both a blackboard architecture and an object oriented environment for implementing our world model framework. Blackboard architectures [13] offer many appropriate properties as a basis for implementing the world model. They provide a mechanism for separating procedural specification from declarative knowledge, the flow of control is determined at run time by a scheduler according to the state of the blackboard (ie what data and hypotheses have been generated), and multiple and/or partial hypotheses can be asserted and retracted. Although blackboard systems can employ frames [14], they are most appropriate where there is less prior knowledge than we expect to have in the context of TUPIP. Furthermore, although the architecture itself is versatile and domain independent, rudimentary generic scheduling rules normally have to be augmented by application-specific mechanisms.

Object-oriented programming would provide direct support for some aspects of a frame-based system, but does not cater for functions which are not methods of specific frames.

We are implementing our frame system using Inference ART† which allows flexibility in the design both of the representational structures and of separate rule bases used to implement functions such as spatial reasoning and the control mechanisms.

Many of the instantiation methods associated with model elements involve procedural algorithms which it would be inappropriate to implement at the level of frame representation. Equally, these algorithms must directly manipulate image and other data which do not need to be directly represented or made explicit at the level of the world model. In our system such methods are implemented in Pascal and may use special purpose hardware. They are therefore grouped together with the relevant data structures into modules, which are objects in the sense of object-oriented programming, so that they can be invoked during instantiation of the world model via a message passing mechanism. The messages are procedurally attached to slots of the relevant frames, where such slots usually represent symbolic references to the data structures within the modules. This allows the correct degree of control over low-level processing despite the separation between the symbolic and numeric levels of computation.

EXAMPLES

We describe how the above framework can be applied to two visual tasks of a contrasting nature, both of which have been the subject of previous study leading to specific solutions against which the performance of the proposed system can eventually be compared. The following examples demonstrate our intended use of the system and serve to illustrate the above description.

Chromosome Classification

Figure 1 shows the genetic material from a human cell arranged, as it is during cell division, into a set of chromosomes which appear as darkly stained bodies. In order to detect genetic abnormalities it is necessary to identify individual chromosomes on the basis of size, shape, and pattern of stain uptake (banding pattern). Some of these properties have to be measured relative to a particular image. The desired result is a display called a karyogram where identified chromosomes appear in a fixed format as shown in figure 2. A semi-automated interactive system which addresses this task is described by Graham [15].

Figure 3 shows the model structure for this example. Figure 4 shows the relevant portion of an inheritance

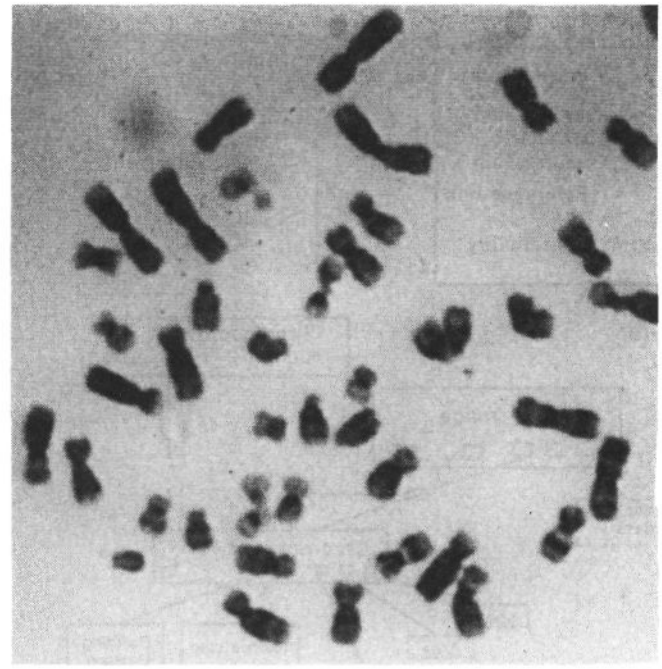

Figure 1. A cell image showing a set of chromosomes.

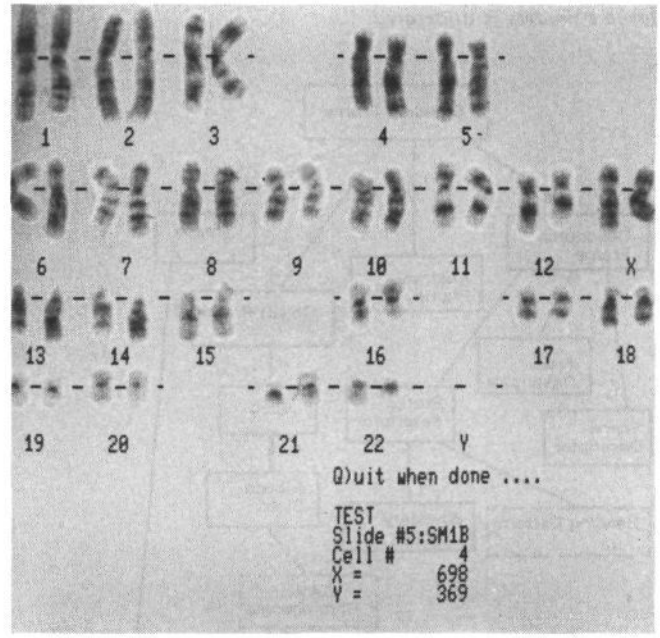

Figure 2. A typical karyogram display

hierarchy indicating how elements of the model in figure 3 have been specialised from generic prototypes. The karyogram can be considered as a specialisation of a cell image in which the positions of chromosomes are exactly constrained, and can thus be generated from an actual cell image whose chromosomes have been identified.

The major problems posed by this example are in resolving ambiguities due to touching or overlapping chromosomes and in producing an optimal classification given missing data due to obscured parts of banding patterns and boundaries. No spatial constraints exist and there is a great deal of variability in the appearance of chromosomes. This variability means that individual blobs in the cell image might match best to the

† ART is a trademark of the Inference Corporation of Los Angeles

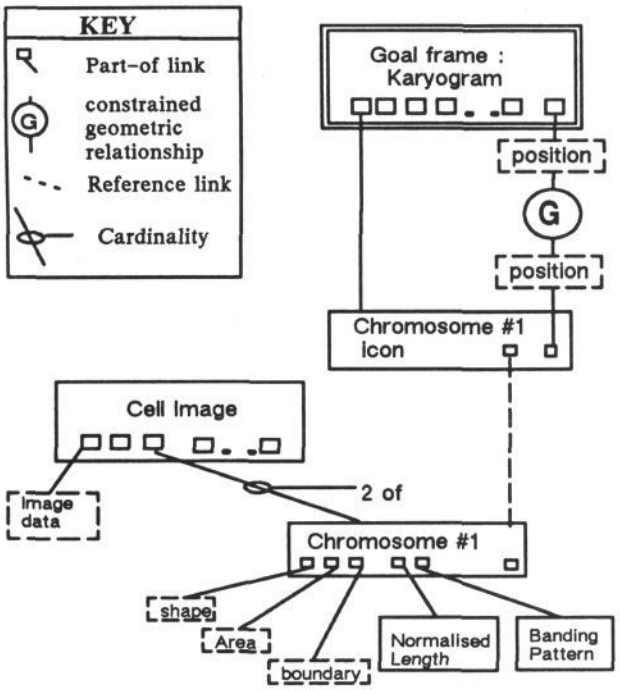

Figure 3. The world model for the chromosome analysis example. Boxes with dotted outlines represent model elements automatically provided by inheritance. Only one of 25 chromosome models is illustrated.

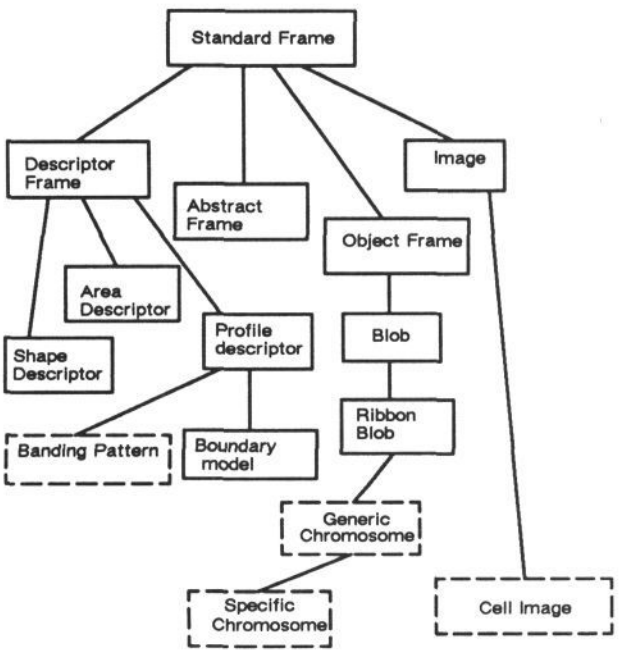

Figure 4. Portion of the inheritance hierarchy showing user defined model elements (dotted boxes), and system prototypes (solid boxes), for the chromosome classification example.

incorrect chromosome model. The system might therefore be expected on occasion to instantiate too many instances of some chromosomes and too few of others, but the model makes such errors explicit. Our control structure must be able to use this information to focus further processing toward re-interpretation of the 'excess' candidates as instances of the missing ones.

Inspection of Complex Assemblies

In this example the goal is to detect faults in a complex mechanical assembly, specifically a car rear drum-brake as shown in figure 5, on the basis of a set of

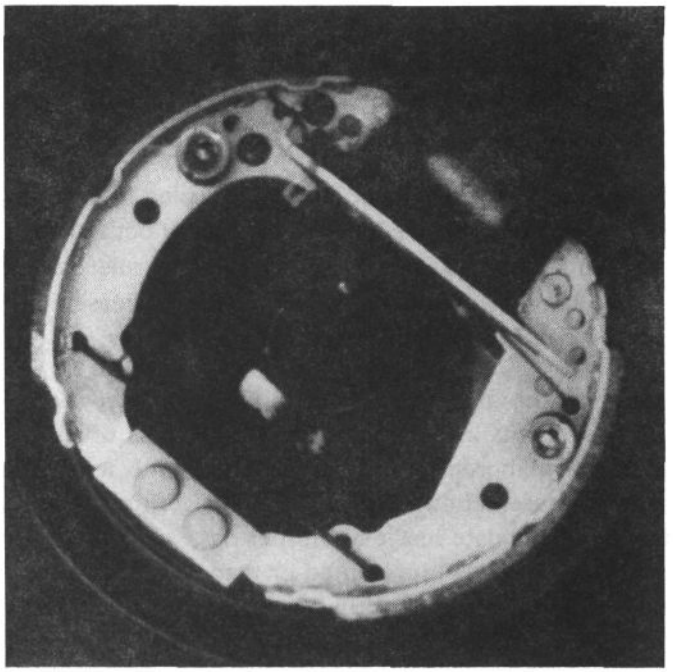

Figure 5. One view of a brake assembly.

user-defined tests. These involve determining that components are present, correctly fitted and undamaged. The world model is complex but constrained, allowing well defined geometric relationships between sub-parts to be exploited. A model-based but procedural approach to this problem is described by Woods *et al* [10]. For clarity we discuss here a simplified case where only one measurement is used.

Figure 6 shows the the world model and figure 7 shows the relevant portion of the specialisation hierarchy. The

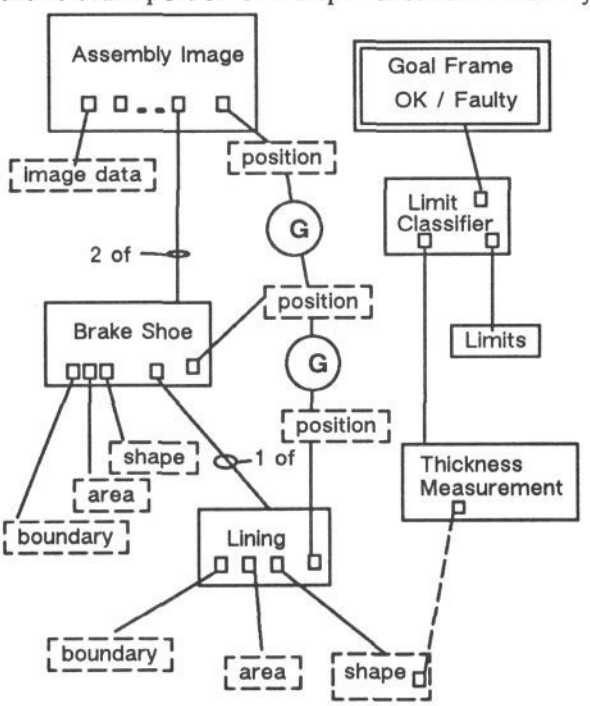

Figure 6. A simplified world model for the industrial assembly example. See key in figure 3.

user elects to use a limit classifier to test that the lining thickness is within an allowed tolerance. The classifier requires a real number derived in this case from the

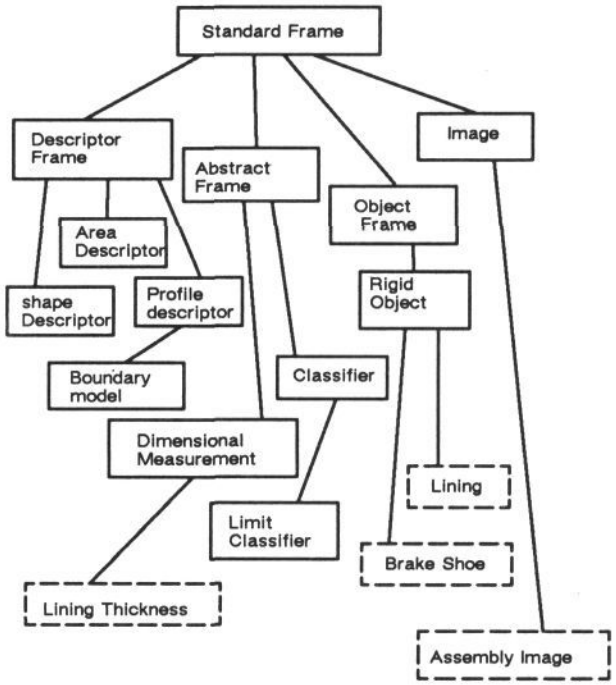

Figure 7. Portion of the inheritance hierarchy showing user defined model elements (dotted boxes), and system prototypes (solid boxes), for the assembly example.

instantiated lining shape description. User interface methods associated with the dimensional measurement frame with a fixed vocabulary of terms such as thickness, would allow the user is to establish this link without understanding the internal shape descriptor.

The lining must become instantiated for the goal to be achieved but this might involve first finding the shoe to which it is attached. In general, some components will be easier to locate unambiguously than others and so the most robust instantiation strategy might involve objects such as the shoe which are not essential for the goal. This provides a good test of the aspects of control that involve the use of importance and utility factors.

REFERENCES

1. **Rao, A R and R Jain.** 'Knowledge Representation and Control in Computer Vision Systems' *IEEE Expert. Spring 1988* pp 64.
2. **Minsky, M.** 'A framework for Representing Knowledge' in *'The Psychology of Computer Vision'* , ed. P Winston. McGraw Hill (1975).
3. **Kuipers, B.** 'A frame for frames: Representing knowledge for recognition', in Bobrow and Collins (Eds) *'Representation and Understanding'*, Academic Press, pp 151– 184.
4. **Rich, E.** *'Artificial Intelligence'* McGraw Hill pp 229–233 (1983).
5. **Hanson, A and E Riseman.** 'VISIONS: A computer system for interpreting scenes' in *'Computer Vision Systems'*, (eds) Hanson and Riseman. Academic Press (1978).
6. **Hanson, A and E Riseman.** Univ. of Massachusetts at Amherst COINS Internal Report 87–05 (1987).
7. **Tsotsos, J K.** 'Representational Axes and Temporal Cooperative Processes' in *'Vision, Brain, and Cooperative Computation'* (eds) Arbib and Hanson. MIT Press (1987).
8. **Tsotsos, J K.** 'Knowledge Organisation and its role in Representation and Interpretation for Time varying Data: the ALVEN system.' *Comput. Intell.* Vol. 1 pp 16 (1985).
9. **Cooper, D C, N Bryson and C J Taylor.** 'An Object Location Strategy using Shape and Grey-level Models' in *'Proceedings of the Alvey Vision Conference'*, Manchester, 1988.
10. **Woods, P W, C J Taylor, D H Cooper and R N Dixon.** 'The use of geometric and grey-level models for industrial inspection'. *Patt. Rec. Letters* Vol 5 pp 11 (1987).
11. **Thornham, A, C J Taylor and D Cooper.** 'Object cues for model based image interpretation' in *'Proceedings of the Alvey Vision Conference'*, Manchester, 1988.
12. **Graham, J, and C J Taylor.** 'Boundary cue operators for model based image processing' in *'Proceedings of the Alvey Vision Conference'*, Manchester, 1988.
13. **Nii, H Penny.** 'The Blackboard Model of Problem Solving and the Evolution of Blackboard Architectures' *AI Magazine* Vol 7 pp 38–53 (1986). and 'Blackboard application systems and a Knowledge Engineering Perspective' *AI Magazine* Vol 7 pp 82–106 (1986).
14. **Towers, S.** *'Frames as data structures for SBS'*. MRC report (unpublished) 1987.
15. **Graham, J.** 'Automation of routine clinical chromosome analysis I. Karyotyping by machine' *Analyt. Quant. Cytol. Histol.* Vol. 9 pp 383–390, (1987).