USING GRAMMARS FOR SCENE INTERPRETATION

H.I. Christensen

Laboratory of Image Analysis
Aalborg University
Fr. Bajers Vej 7, Bldg D1
DK-9220 Aalborg East
Denmark

J. Matas & J. Kittler

Vision, Speech & Signal Proc. Group
Dept of Electrical and Electronic Engr.
University of Surrey
Guildford, GU2 5XH
United Kingdom

ABSTRACT

A method that employs grammars to direct the inference process of a vision system that does interpretation of dynamic scenes is described. The system uses a set of qualitative image descriptors to drive the interpretation. The result is a 'natural language' description of scene activities. In addition the inference engine generates a set of predictions that can be used to control the interpretation strategy so as to make the processing of new images more efficient. The system has been implemented in an expert system shell to demonstrate the viability of the approach. Results on real images are reported.

1. INTRODUCTION

It is well known that the world is, to a large extent, structured in space and time. This is for example exploited in spatial image analysis for line extraction, feature grouping and in geometry based object recognition. The structuring in time is used in motion analysis to support the assumption about the smoothness of motion, bounded dynamics etc. These example are all 'image level' manifestations of spatio-temporal structures. At the scene level the evolution is often guided by laws (i.e. traffic), by social conventions (i.e., how we sit on chairs), or by traditions (i.e., placement of fork and knife on a table). The evolution / dynamics of individual objects and their relationship is typically sequential by nature. Such sequences of actions can be captured in a language.

Many have reported on methods for dynamic scene description in terms of a 'language'. Good examples include [2, 5, 3, 6]. These approaches rely on the tracking of objects to generate object trajectories. These trajectories are then partitioned into segments that are assigned semantic labels. The partitioning of trajectories is driven by a combination of spatial position and trajectory characteristics like curvature. A notorious problem in the tracking based approach is robustness. Trackers loose targets temporarily due to noise and occlusions. To achieve a robust interpretation performance it is suggested that interpretation should possess two characteristics: a) it should be driven by qualitative features and b) it should have facilities for error-recovery. In addition it is desirable that the interpretation system can generate predictions which can control low level processes.

In this paper we describe a grammar based approach to interpretation and describe how it may be driven by qualitative features. The system has been implemented as a rule-based expert system. We will demonstrate the performance of the system for interpretation of a table setting scenario.

This work was sponsored by the EU Long Term Research project "Vision as Process" EP-7108-VAP II.

2. SCENE GRAMMAR AND INFERENCE ENGINE

Scene evolution can typically be captured by sequential expressions, that can be described in BNF form, e.g.:

```
<TEA_BREAK>::=<SETTING>,<DRINK>,<CLEANUP>
<SET_A_CUP>::=<SET_SAUCER>,<SET_CUP>
<SET_CUP>  ::=enter-fov(cup),align(cup,saucer),
             align(cup,table), static(cup)
```

These expressions demonstrate that the description of evolution must include the following characteristics: i) should be based on observable features, ii) must describe individual objects, iii) must encompass relations between objects and iv) include compositions of actions.

Such descriptions can be captured in a regular grammar, represented by the generator $\mathcal{G}$:

0-7803-3258-X/96/$5.00

$$\mathcal{G} = (Q, \Sigma, P, q_0, Q_m)$$

where

Q is the set of states, or steps in the interpretation.

Σ is the set of features that drive the interpretation. By nature these features must be discrete.

P is the set of productions that describe the evolutions from one state to another, given a particular feature is detected. $(P \subseteq (Q \times \Sigma \times Q))$.

q_0 is the initial state which is the entry point for the interpretation procedure.

Q_m is the set of terminal or marker states $(Q_m \subset Q)$, which indicate that the interpretation of a 'phenomenon' has been completed.

The states (Q) and the productions (P) are related both to simple actions and to compositions. It is necessary to capture both in order to facilitate scene level interpretation, which is a multi-scale process. The interpretation is driven by the features (Σ), that by their very nature are qualitative and discrete. The features are detected by a small set of dedicated image processing procedures. Examples of features include 'recognised objects' (i.e., cup & saucer), 'geometric relations' (i.e., aligned & parallel), 'temporal discontinuities' (i.e. enter-field-of-view & static), etc.

Once a 'feature' (σ_i) is detected and given the present state (q_i) a production (P_i) is invoked which results in a new state (q_j). The actual transition from state $q_i \rightarrow q_j$ denotes a step in the scene interpretation. Each transition in P has thus an associated semantic description that describes the 'machine' or 'operator' interpretation of the observed phenomena. In addition each transition has an associated action specification $\mathcal{A}$ that enables execution of specific actions. For interpretation in a production environment the action specification could specify manipulation or initiation/termination of a particular manufacturing process.

Handling of errors originating from the image processing is a well known problem in scene interpretation. To achieve robustness in the interpretation it is necessary to incorporate error recovery into the process, so that the system can cope with missing or incorrect features. Error recovery can be addressed by using sub-string matching. The grammar, $\mathcal{G}$, defines a language, $\mathcal{L}$. By using concatenations of features the result can be matched against legal constructions in the language. E.g., in the example above an unrecognised object may enter the field of view and be placed on a saucer. By sub-string matching (i.e., analysis of future productions), it is possible to hypothesise that the object is a cup.

At any of its stages the status of the interpretation process is captured by the present state q_i. In such a state only a subset P_i of the productions is relevant $(P_i = P \cap (q_i \times \Sigma \times Q))$. The productions P_i also define a set of events (Σ_i) that correspond to the events that are expected to occur in the scene. This set of expected events can be used for control of the low-level modules. That is, the events combined with contextual information, derived from the participating states, enable selective/purposive image analysis. In the example above an expectation is the feature `'align(cup,saucer)'`, which is a specific geometric relationship, that can be checked by a 'simple' analysis routine, applied in a local neighbourhood of the saucer. To ensure detection of unexpected events and to enable recovery from errors the purposive routines should always be complemented by a pre-attentive event detection module.

Grammars are usually interpreted by parsers, as described in [1]. In this context the interpretation must proceed at all scales of the scene concurrently, i.e., all of the productions in the example above might be relevant at the same time. There is thus a need for a set of concurrent parsers. In addition the parsers must be able to perform error recovery through sub-string matching. To accommodate these requirements the parser has been implemented in a rule-based expert system. The rule-base contains a generic parser, that can parse multiple strings concurrently. The specific grammar for a particular domain is encoded in declarative knowledge (a graph representation of patterns). For the purpose of interfacing to external agents each production (p_i) has an associated 'action specification', that may be used for control of actuators, as outlined above.

Expert systems are typically thought of as systems based on heuristic information with poorly understood processing characteristics. In our system this is not true as the processing is driven by the grammar, which has well defined properties. The grammar may also be used for formal verification of the system, which is highly desirable.

3. AN EXAMPLE

To demonstrate the utility of our approach the system has been evaluated on a 'tea-drinking/table-setting' scenario. A few images from the domain are shown in figure 1.

The inference is carried out by the generic parser with is implemented in terms of a rule base with 42 pro-

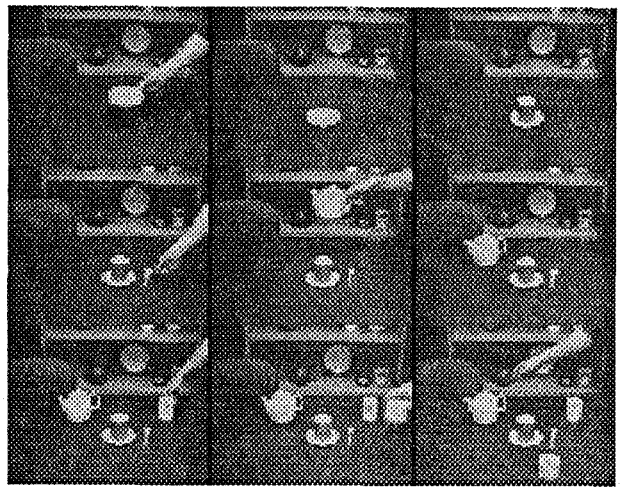

Figure 1: A few example images from a sequence that is used for demonstration of the utility of approach described. Colour and simple geometry is used for recognition.

duction. The specific characteristics/evolution of the test domain is encapsulated as declarative knowledge that is encoded in 60 facts (action patterns). Due to space limitations, the rules and the declarative knowledge cannot be reproduced here.

The qualitative image features that drive the interpretation are simple qualitative image descriptors, as described below:

enter-fov(X) The border of the image is searched for new regions that might indicate the presence of a new object. If a region is found, colour and geometry is used for classification, see [4] for details. The event is thus a signal of the presence of object X in the field of view. In the present system X$\in$ (cup, saucer, pot, spoon, sugar-bowl, milk-jug, unknown). The event returns both the generic class and a unique object id, to allow the presence of several objects from the same class in the image at the same time.

left-fov(X) This feature indicates that the object X has left the field of view.

place(X,Y) This is a geometric grouping procedure that detects that object, X, has been placed on object Y and now no motion is associated with object X.

static(X) Indicates no motion is associated with object X.

moving(X) Indicates that object X is moving.

align(X,Y) Indicates that objects X and Y are aligned in the (x, y) coordinate system defined by the tabletop. This is for example used for the detection of pouring of tea.

motion(X) This indicate that a 'stationary motion pattern' is associated with object X. Stationary motion patterns are detected by the Fourier analysis of the trajectories. The feature is used for the description of movements related to stirring.

The rule base and the grammar has been implemented in the rule based expert system system CLIPS from NASA[7], while the image processing is carried out by dedicated procedures implemented in C.

The system has been evaluated on sequences as those shown in figure 1. Output from the interpretation is shown below. Please note that there is not a one-to-one correspondence between the images shown in 1 and the output. Due to the space limitations it is not possible to show the entire sequence.

```
Frame 0  saucer_1 has entered FOV
Frame 1  saucer_1 has been placed on the table
         saucer_1 (enter-fov,place) ->
         Sub-plan set-saucer completed
Frame 3  cup_1 has entered FOV
Frame 5  cup_1 has been placed on the table
         cup_1 (enter-fov,place) ->
         Sub-plan set-cup completed
         (set-saucer,set-cup) -> set-a-cup
Frame 7  spoon_1 has entered FOV
Frame 9  spoon_1 has been placed on the table
         spoon_1 (enter-fov,place) ->
         Sub-plan set-spoons completed
Frame 11 teapot_1 has entered FOV
Frame 13 teapot_1 has been placed on the table
         teapot_1 (enter-fov,place) ->
         Sub-plan set-pot completed
Frame 16 milkjug_1 has entered FOV
Frame 17 milkjug_1 has been placed on the table
         milkjug_1 (enter-fov,place) ->
         Sub-plan set-milkjug completed
Frame 20 sugarbowl_1 has entered FOV
Frame 21 sugarbowl_1 has been placed on the table
         sugarbowl_1 (enter-fov,place) ->
         Sub-plan set-sugarbowl completed
         (set-spoons,set-milkjug,set-sugarbowl)->
         set-aux
         (set-a-cup,set-aux,set-pot) -> setting
```

As mentioned in section 2 each production has an associated action specification. In this particular example the action specification is used for output of a 'natural-language' like description of the activities in

the scene. The text description corresponding to the above interpretation is listed below

```
F( 0) A saucer has been put on the table
F( 5) A cup was placed on the table
F( 5) A cup with saucer has been placed on table
F( 9) A spoon was placed on the table
F(13) The tea pot is now on the table
F(17) The Milkjug is now on the table
F(21) The sugarbowl is on the table
F(21) The tea-break auxiliaries are on the table
F(21) The table has been set, 'Tea is served!'
```

4. SUMMARY/DISCUSSION

In this paper we have argued that the interpretation of dynamically changing scenes may be based on a simple grammar representation, driven by qualitative scene features that can be robustly extracted from natural images. The approach includes facilities for recovery from errors, and control of low level processes. To demonstrate the utility of the approach, results obtained on a sequence of images of a natural scene have also been presented.

The developed system consists of three components: i) a generic parser for the interpretation, ii) a set of productions that encode the relationship between interpretations and simple sequences of qualitative image features, and iii) a set of image processing routines for detection of qualitative scene features.

The generic parser is general and can be used for a large variety of domains. The qualitative image features are computed by recognition routines or are obtained by methods for identification of feature groupings, and detection of qualitative temporal phenomena. This part is thus general in nature even though the recognition routines must be tailored to specific applications. Finally the productions are specific to the application domain, but as it is a separate component it can easily be changed.

In consequence a general framework for grammar based interpretation of dynamic scenes has been presented. Through minor modifications to the system it is possible to apply the same framework in a variety of applications.

Future work will emphasise in particular the use of the same system on a variety of domains to demonstrate the claimed generality of the proposed approach.

5. REFERENCES

[1] A. Aho, R. Sethi, and J. Ullman. *Compilers, Principles, Techniques and Tools.* Addison Wesley, 1986.

[2] N. I. Badler. *Temporal Scene Analysis: Conceptual Descriptions of Object Movements.* PhD thesis, Dept. of Computer Science, University of Toronto, Canadian Thesis on Microfische 33080, 1975.

[3] R. J. Howard and H. Buxton. An analogical representation of space and time. *Image and Vision Computing,* 10(7):467–478, September 1992.

[4] J. Matas *Colour-based Object recognition* PhD Thesis, Dept of Electrical and Electronic Engineering, Univ of Surrey, Guildford, UK, 1995.

[5] H.H. Nagel. From image sequences towards conceptual descriptions. *Image and Vision Computing,* 6(2):59–74, May 1988.

[6] B. Neumann. Natural language descriptions of time-varying scenes. In D. L. Waltz, editor, *Semantic Structures: Advances in Natural Language Processing,* pages 167–206. Lawrence Erlbaum Associates, 1989.

[7] G. Riley. *CLIPS Programmers Manual.* NASA Johnson Space Center, Houston, TX, 6.04 edition, March 1994.