Proceedings of the Sixth International Joint Conference on Artificial Intelligence, 1979, Tokyo, pp.684-686

# A PRODUCTION SYSTEM FOR REGION ANALYSIS

Yu-ichi Ohta, Takeo Kanade, and Toshiyuki Sakai

Department of Information Science
Kyoto University
Kyoto, 606, JAPAN

Production system architectures are useful for knowledge representation. We have applied the architecture to image analysis in the framework of region growing and developed an outdoor-scene analyzer. In this paper, the following three problems are addressed to make production systems workable in scene analysis. 1) Appropriate size of knowledge represented in a production rule. 2) Reduction of computation. 3) Control of the analysis toward goal.

## 1. INTRODUCTION

A production system consists of a set of production rules and a database. A production rule is the unit of knowledge representation and the database records the facts about input image. The control structure of production systems is heterarchical. Each production rule is a pair of a condition and an action and is "watching" the database. Whenever the predicate in the condition part is satisfied, the system evaluates the action part and modifies the database. Because of this, it is easy to add or modify knowledge in production systems. This is a useful feature to organize an analysis system for complex scenes, such as outdoor scenes, which include various kinds of objects.

We have made an outdoor-scene analyzer in the framework of region growing using production system for knowledge representation and control structure[1]. In this paper, we describe the outline of our scene analyzer and discuss production systems from the view point of image analysis.

## 2. OUTLINE OF ANALYSIS MECHANISM

Figure 1 shows the schematic diagram of the analysis mechanism in our system. The system receives an input color image as digitized red-green-blue intensity arrays and constructs a semantic description of the scene.

Preliminary segmentation - The primary objective of the preliminary segmentation process is not the reduction but the structuring of raw image data into usable information; The input color image is segmented into a set of coherent "patches" and organized into a fully structured symbolic description. In the higher-level processes, all picture-processing operations are performed on this structured description rather than the raw image data. The "patch" is an important element to describe the input image and to build a production system in our region analyzer.

Plan generation - The plan generation process extracts the overall structure of the scene to obtain clues concerning which knowledge should be applied to what part of the scene. First, patches with large area are selected as key-patches from the segmented image. It is reasonable to assume that most of them correspond to large parts of objects in the scene. Labels of objects are assigned to each of the keypatches and the degree of correctness is computed using

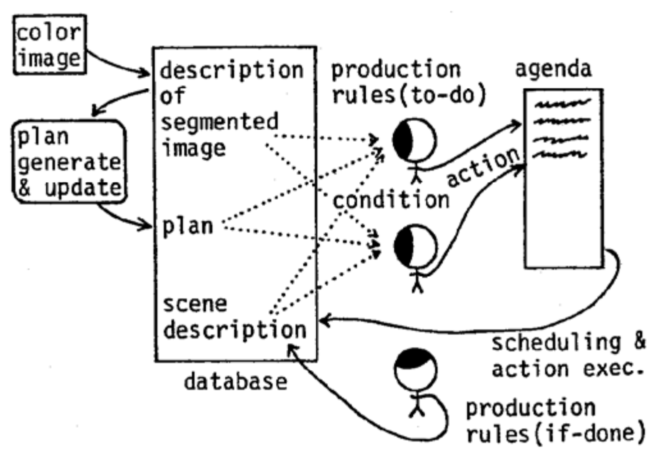


Fig. 1. Outline of the Analysis Mechanism

knowledge, which is represented as a collection of knowledge-blocks[2], about the properties of and the relations between objects. The result of the interpretation, called a plan, is a set of labels and their correctness values assigned to each keypatch, e.g. (sky=0.6, building=0.2, tree=0.1, road=0.1), etc. The plan is updated whenever significant fact, such as the position of scene horizon, is found by the production rules in the analysis process.

Database - The structured description of a segmented image, the plan, and the scene description so far obtained are all stored in the database. Figure 2 illustrates the structure of the scene description built as the result of analysis. Scene, object, region, sub-region, patch, and pixel are the important concepts which constitute the hierarchical structure of our description. The patch corresponds to the result of the preliminary segmentation. All the descriptive elements are organized into the hierarchical structure by "part-of" relation. Relations between parts of objects, such as "adjacent" or "occluded", are described between corresponding regions.

Production rules and agenda - There are two types of production rules in our system; to-do rules and if-done rules. They correspond to consequent and antecedent theorems of PLANNER[3],

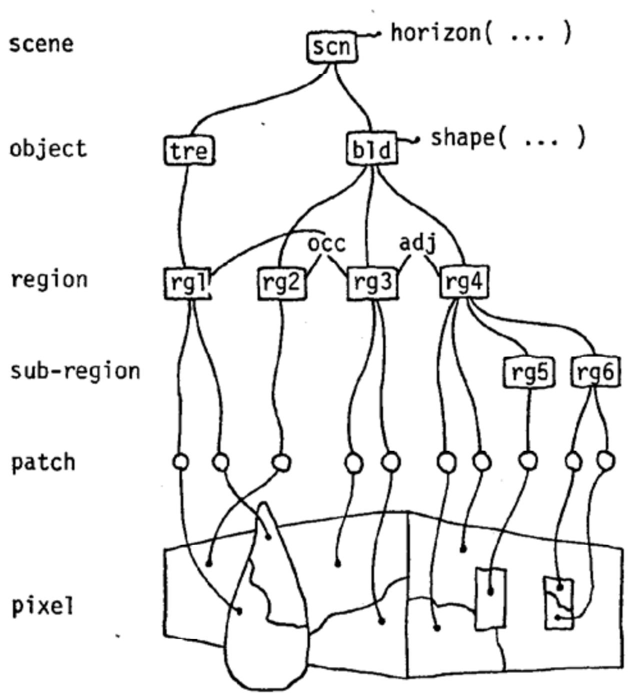


Fig. 2. Structure of the Scene Description.

respectively. A to-do rule performs basic operations in region growing process. It examines each patch which is not yet interpreted in the segmented image by the fuzzy predicate in its condition part and determines whether the action can be executed for it. The executable action is given a score to indicate its priority, and added into the agenda. The agenda manages those actions which are executable in the present context through their scores. The action with the highest score is executed and as the result the agenda is updated. An if-done rule is triggered by the execution of a certain action of the to-do rules. It performs an additional operation in making the scene description, such as extracting the position of scene horizon, etc.

## 3. DISCUSSIONS

We have to solve three problems to import the production system architecture to image analysis. The first is the problem of what "size" of production rule is appropriate for region analysis, the second is the problem of how to reduce the amount of computation, and the last is the problem of how to direct the analysis to goal.

### ° What "size" of production rule is appropriate?

What size of knowledge should be represented by one production rule? As an example of "large" knowledge size, it is possible that each production rule corresponds to one object to be extracted and has whole knowledge about the object. This scheme enables to perform skillful analysis according to the characteristics of each object. On the other hand, the rules become large and complex and it is difficult to manage them.

Small sized knowledge - In our system the size of knowledge represented by a rule is fairly small. Each rule is described as a combination of basic operations in region growing: selecting an un-interpreted patch from the segmented image, assigning a label to it, and assembling it into the scene description. This scheme has the following merits: each rule is simple and easy to modify, and the interaction among rules can be performed in a clear way because the access method to the database is simple.

Reinforcements - Because the analysis proceeds patch by patch in our scheme, it is difficult to deal with global constraints such as object shape or relation between objects. We took three steps to cope with this situation. First, a plan is generated as rough interpretation of the input scene. This enables the production rules to catch information about global structure of the scene. Secondly, patches can be dealt with as a set at a time, and it becomes possible to extract object which is defined as a combination of mutually constrained patches such as windows of a building or a car on road. Lastly, we have devised special rules which extract information from the segmented image without sticking to the patch-by-patch analysis. This kind of rules, for example, is used to extract the shape of a building.

° How to reduce the amount of computation?

In production systems the control program must grasp the state of executable actions at any moment in the analysis process. The agenda must be updated whenever the database is changed. Roughly speaking, the number of tests to be done each time is estimated as:

(the number of un-interpreted patches)
X (the number of production rules).

It becomes several thousand and it is too many. Furthermore, the predicate in the condition part of the production rules includes time-consuming picture-processing functions. It is necessary to reduce the number of the patches and production rules which must be actually examined at a time. For this, the structure of scenes must be taken into consideration.

"Globality" and "Locality" - A scene usually has two different properties from the view point of image analysis: "Globality" and "Locality". Analysis results such as determination of scene horizon or detection of objects may have significant influence on the analysis of overall structure of the scene. This property is called "Globality". On the other hand, the results of analysis in a small part of an object scarcely has influence on the analysis of other parts in the scene. This property is called "Locality". These two properties are utilized in the control structure of our scene analyzer.

Scene phase and object phase - In order to deal with such two properties of scenes, two phases are set up in the control program. They are scene phase and object phase. The task of the scene phase is to analyze overall structure of input scene without sticking to details. So, it is almost meaningless to examine small patches in the scene phase analysis and only the key-patches are examined. Whenever a keypatch is labeled, the scene phase is activated and all un-interpreted keypatches are re-examined. In the object phase, analysis of detailed structures are performed under the context of the results in the scene phase analysis. When a patch which belongs to a object is labeled, the object phase corresponding to the object is activated and the patches touching the patch just labeled are examined or re-examined.

Dividing rules into subsets - The set of production rules can be divided into subsets to be used in the scene phase and the object phases corresponding to each object. In each phase, only the subset corresponding to that phase is activated by the control program to examine the un-interpreted patches.

Consequently, the number of tests to be done at a time is reduced to several 10's.

° How to direct the analysis to goal?

It is an important feature of production systems that each production rule independently checks the database and modifies it whenever the condition is satisfied. However, two problems need to be considered to make this mechanism actually work; a method to direct the analysis to goal and a method to resolve the conflict among the executable actions which are inconsistent with each other.

Score - As was described, every executable action is registered in the agenda with its score and the action which has the highest score is executed to change the database. Then the score plays an important role to direct the analysis toward goal. The value of score associated to each executable action is calculated as the sum of a base value and a premium value. The base value is a constant given to each production rule, and it plays a role to specify the order of analysis. The premium value is given as the degree of satisfaction of the condition of production rule at the situation the rule was examined, and it plays a role to guide the analysis toward the correct interpretation. To sum up, the analysis proceeds toward goal guided by the premium values following the strategy specified by the base values.

Conflict resolution - In our system, the basic operation which changes the database is very simple: to assign an object label to a patch. This makes the detection of the inconsistent actions quite straightforward. Whenever a patch is interpreted by executing an action registered in the agenda, every action which is going to give a different interpretation to the patch just interpreted is decided to be inconsistent and deleted from the agenda.

In this paper, we described a production system which was applied to region analysis. Some problems and their solutions were discussed.

REFERENCES

[1] Ohta,Y. et al. "An Analysis System for Scenes Containing Objects with Substructures." In Proc. 4th IJCPR, November, 1978, pp.752-754.
[2] Sakai,T. et al."Model-based Interpretation of Outdoor Scene." Proc. 3rd IJCPR, Nov. 1976.
[3] Hewitt,C. "PLANNER" A.I.Memo No.168, MIT, October, 1968.