AN ANALYSIS SYSTEM FOR SCENES CONTAINING OBJECTS WITH SUBSTRUCTURES

Yu-ichi Ohta, Takeo Kanade, and Toshiyuki Sakai

Department of Information Science
Kyoto University
Sakyo, Kyoto, 606, JAPAN

SUMMARY

This paper describes a scene analysis system which can perform semantic segmentation of a scene with hierarchical structure. An input color image is first segmented into a structured symbolic description using intensity data. Semantic interpretation is performed on this description. Top-down control and bottom-up control are combined in the framework of semantic region growing. Knowledge is represented as a semantic network in the bottom-up process and in the top-down process as a set of production rules. The system was applied to outdoor scenes.

1. INTRODUCTION

It will be easily recognized that almost all scenes have hierarchical structure. It is an overall/detail hierarchy[1] which is formed by the fact that each object in the scene has substructures. In a town scene, for example, sky, roads, and buildings make the overall structure of the scene as the first-level objects in the hierarchy, and windows of the buildings and cars on the roads constitute the detail of the scene as the second-level objects. This paper describes a mechanism to analyze such scenes in the region analysis paradigm.

The region growing technique is often employed to segment a scene into regions. Outdoor scenes are too complex to be segmented into meaningful regions by using only the spectral information. Yakimovsky[2] has incorporated semantic information in the region growing process and succeeded in analyzing road scenes. Yakimovsky's technique, however, cannot cope with scenes which contain objects with structure such as buildings. He entirely relies on bottom-up control in his segmentation program and the semantic information is used in the same way as the intensity difference between regions; it is used only as a criterion to erase boundaries. Even though the segmentation is evaluated by globally optimizing a score, the properties which can be used are limited to local ones; color, orientation of a boundary segment, crude shape of a boundary segment, etc. Such a problem is caused intrinsically by the bottom-up control scheme.

In order to save the situation, employment of model-driven top-down approach seems essential. Under the top-down control it becomes possible to deal with the hierarchical structure in the scene and to deal with a set of regions located separately in the image. The top-down scheme, of cource, has a lot of defects and our aim is to combine the merits of the bottom-up and top-down scheme in the framework of region growing.

2. PRELIMINARY SEGMENTATION OF COLOR PICTURE

Figure 1 shows the flow of data in our system for semantic region segmentation. The part enclosed by a dotted line is performed under bottom-up control. Input color image is first segmented into a set of coherent patches according to the spectral data (see fig. 5-a and 5-b). We employed the recursive thresholding technique[3] for the segmentation. It uses histograms of some color features to determine the cut-off value of thresholding operation for region extraction. We use only three color features, R+G+B, R-B, and (2G-R-B)/2, to compute the histograms. We have got a result that satisfactory segmentation can be performed using these few color features[4].

Using histograms to find cues for segmentation, the recursive thresholding technique is suited to extract rough structures in the image, but is poor at extraction of detail. This weak point is improved by the following technique. When no valley is found on the histograms computed for a large part of the image, the area is scanned by a small rectangle window and histograms are derived from the window at every position. Trials are made to find valleys on each of them. Thus our segmentation program can find detailed structures and extract sufficient information, from rough to detail, out of the input color image.

The segmented image is described in a well organized data structure. Patches, boundary segments, vertices, holes, and (straight) line segments are used as descriptive elements. Parameters and pointers are described in each element to represent properties of and topological relations between them.

When the analysis process is performed by means of a top-down strategy, it is necessary to retrieve patches out of the image data by specifying the property they should have; for example, "fetch all patches which are vertically long and have a yellow patch on their left". Only a well structured symbolic description can allow such a function to be of practical use. The following three primitive functions are prepared for the purpose.

ALL-FETCH(to-set, from-set, fuzzy-predicate)
THERE-IS(to-set, from-set, fuzzy-predicate)
T-FETCH(to-set, patch)

ALL-FETCH selects from *from-set*(a set of patches) all patches which satisfies the condition described by *fuzzy-predicate* and assigns them to *to-set*. THERE-IS, existential fetch, selects only one patch first found. Nested use of these functions realizes arbitrarily complicated retrievals. T-FETCH selects all patches which are touching *patch*. This function, of cource, can be realized using ALL-FETCH, but T-FETCH performs faster search utilizing the relational pointers in the structured description of segmented image.

3. PLAN GENERATION BY BOTTOM-UP CONTROL

It is an important problem where to locate the interface between bottom-up process and top-down process. As rough interpretation of the scene, we made a plan by the bottom-up process. The top-down process can contact with the bottom-up process through the plan (see figure 1). The plan has a role to guide the top-down process; which knowledge should be applied to what part of the scene. It is reasonable to consider that most of the patches with large area in the segmented image correspond to the first-level objects in the scene. It should be possible to grasp the rough structure of the scene by assigning the labels of the (first-level) objects to each of the large patches. No definite features are available, the task must be performed by using only pointwise computable features of the patches and ambiguous relational constraint between them. For such a task bottom-up control with global optimization scheme is appropriate.

The patches with large area are selected as key-patches from the segmented image. A plan image is generated by merging all small patches with one of the key-patches (see fig. 5-c). In this operation no semantic information is used. The purpose of generating the plan image is to make the relations between key-patches clearer. Labels are assigned to each patch in the plan image.

The correctness value of each assigned label is first evaluated using properties (without relations). Then they are updated using relations. In the evaluation of relations, the correctness values of the labels at the partner patches of the relations must be considered. They will be also updated by evaluating relations, so a method like the relaxation[5] is employed for the updating operation. It is iterated several times until all correctness values converge. Finally we get a set of labels and associated degree of correctness for each patch in the plan image, for instance (sky=0.6, tree=0.2, road=0.1, building=0.1).

A semantic network is used to represent the knowledge in the bottom-up process as shown in figure 2. Each node of the network is called a knowledge block in our model and it holds knowledge about matter in the world, for instance, object "sky", material "concrete", property "blue", or relation "linear boundary". A chunk of information is stored in the block as a set of declarative rules, including properties it must satisfy and relations it has with other blocks. The rules have the following formats;

property: (type fuzzy-predicate weight),
relation: (type fuzzy-predicate weight FOR label).

The property or the relation represented by a rule is described in the *fuzzy-predicate* which gives a fuzzy truth value. The *weight* indicates the degree of distinguishability of the property or the relation. The *label* specifies the object with which the relation must hold. The way to use those rules is described in [6].

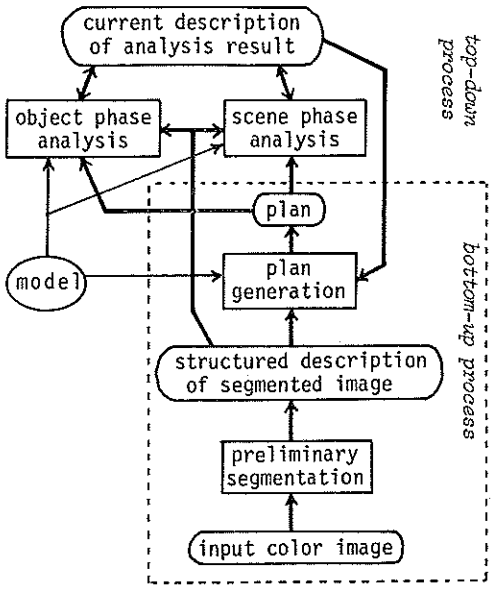


Figure 1. Flow of data in the analysis process.

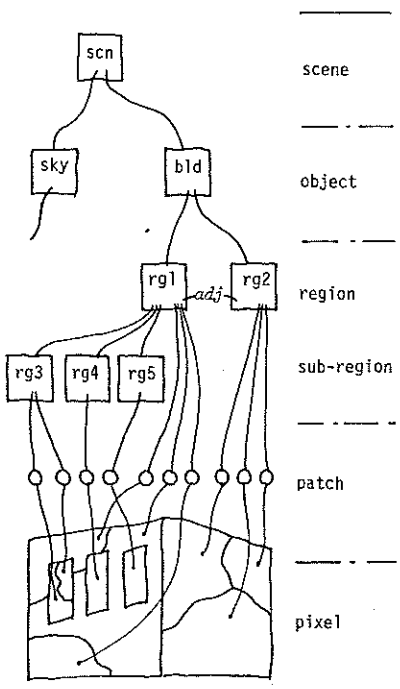


Figure 3. Structure of description built by the system.

4. SEMANTIC SEGMENTATION BY TOP-DOWN CONTROL

All patches obtained in the preliminary segmentation are interpreted in the top-down process referring to the plan generated in the bottom-up process. When the top-down process made a decision such as the position of hirizon which might have a significant effect on the interpretation of whole scene, it is fed back to the plan generation step and the plan is re-evaluated. In this way the top-down process and the bottom-up process work together to achieve the semantic segmentation.

When analysis is performed under top-down control, it is essential to grasp the context of the analysis exactly. Figure 3 shows the structure of the description which is built as the result of the analysis in our system. Scene, object, region, sub-region, patch, and pixel are the important concepts which constitute the structure of our description. The patches are the ones obtained as output of the segmentation process. The regions represent the main parts of objects and are obtained by merging the patches assigned with the same meaning. The objects stand for the objects in the scene. The sub-regions are much the same as the regions but they correspond to the substructures of objects. All descriptive elements are organized into a hierarchical structure by "part-of" relation. Relations between parts of objects such as "adjacent" or "occluded" are described between corresponding regions.

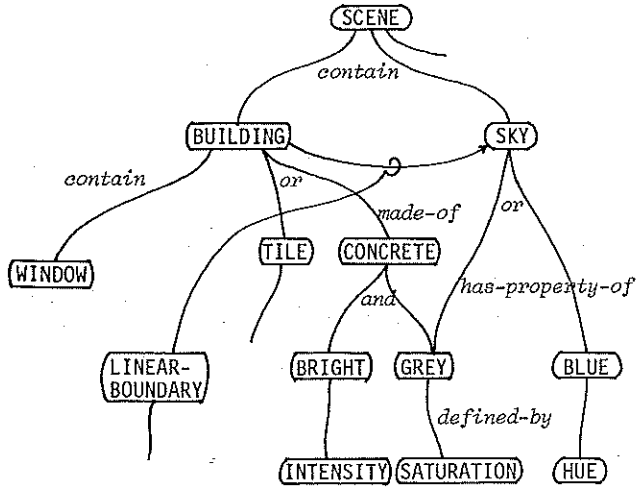


Figure 2. Semantic network for knowledge representation.

The analysis in the top-down process proceeds by iterating simple basic operations; to assign a label to a patch or a set of patches and assemble it/them into the description. At each step in the process, the model and the control structure determine which patch should be interpreted in the current context and how to assemble it into the description. The order of the analysis in the top-down process cannot be predetermined and must be determined depending on the plan obtained in the bottom-up process. So we employed heterarchical control structure and the knowledge is organized as a production system; as a set of production rules. Each production rule has two parts, situation and action, and it accesses and modifies the database which consists of the segmented image, the plan, and the description so far obtained. The situation part is a fuzzy predicate. It examines the un-interpreted patches in the segmented image and decides whether the action part can be executed for the patch. It also gives a priority to the action determined to be executable. Another important role of the situation part is fetch function. It picks up some patches from the segmented image and deliver them to the action part. The action part is a function which assigns an object label to a patch or a set of patches offered by the situation part and assembles it/them into the description.

Let us illustrate a production rule which interpret the windows, a substructure of the building. In order to interpret the windows, it is necessary to assign the label "window" to the all patches which have rectangle shape and are arranged by a rule within the area which

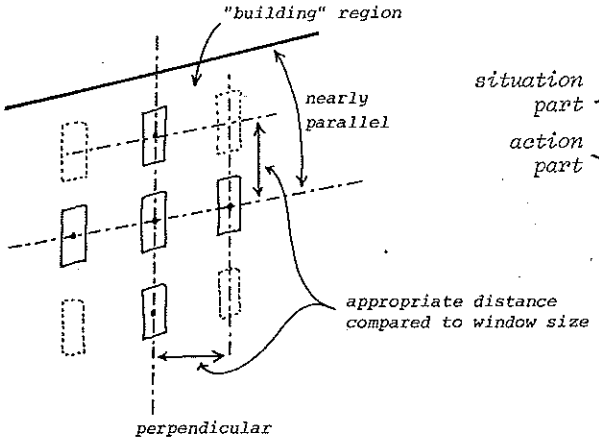


Figure 4-a. "Building" region and "windows".

```
(ACT              "indicates following is production"
(IF (*WINDOW-LIKE *RGN)
   (THEN (GET-SET *PLSET (PLAN *MRGN) PATCHES)
                 "assign to *PLSET all patches which belong to the plan of *MRGN"
         (AND (ALL-FETCH *WLIKE *PLSET
                 (AND (IS (LABEL *WLIKE) NIL)(*WINDOW-LIKE *WLIKE)))
              (ALL-FETCH *WIND *WLIKE
                 (THERE-IS *WK *WLIKE (*W-RELATION *WIND *WK)))
                  "search all patches that have at least one partner patch
                   with which *W-RELATION holds"
              (THERE-IS *WK *WLIKE (IS *RGN *WK)))))
(THEN (CONCLUDE P-LABEL B-WINDOW)   "*RGN is 'window'"
      (FOR-EACH *WIND (MUST-BE *WIND P-LABEL B-WINDOW))
                  "all patches in *WIND must be 'window' if *RGN is 'window'"
      (DONE-FOR *WIND)   "no need to examine the *WIND patches any more
                          under this context --- for controler"
      (PRIORITY (ADD 2100 (NUMBER *WIND))) ))
```

situation part

action part

Figure 4-b. The production for analyzing "windows".

has been interpreted as "building" (see figure 4-a). This operation is described by the rule in figure 4-b. *RGN and *MRGN are specially reserved variables; the former means the patch itself under consideration which may be "window" and the latter the region, a part of "building", to which the window belongs. The meaning of each part of the rule will be clear by the comments in the figure.

All production rules whose actions are determined to be executable are managed by a scheduling table with their priorities. The action with the highest priority is executed and the scheduling table is updated again. At every step in the analysis process, the scheduling table must grasp all production rules whose situations are satisfied in that context. So it must be updated whenever the database is changed. Roughly speaking the number of situations which must be examined each time is estimated by

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

and it becomes several thousand. That is too many. It is necessary to set limits to the patches and production rules which must be examined at a time.

Utilizing the hierarchical structure in the scene, we have employed a strategy which has two phases for the interpretation operation: scene phase and object phase. Decisions about scene structure are made in the scene phase from the global viewpoint referring to the plan. The detail of each object is analyzed in the object phase under the context obtained in the scene phase analysis. In this way the patches which must be examined in the scene phase can be limited to key-patches. In the object phase only the patches adjacent to the patches which have been interpreted already are examined. This limits the patches for which the scheduling table must be updated after each cycle of the interpretation operation to those that are touching the patch newly interpreted in the last cycle.

Having those phases in the control structure, the production rules can be divided into many groups and the number of rules needed to be examined in each phase is reduced. The rules which perform the overall analysis in the scene phase are described in the knowledge block representing "scene", and the rules which perform the detail analysis in the object phase are described in the knowledge blocks corresponding to each "object". The controller can pick up and examine only the effective production rules from the knowledge blocks according to the current phase.

5. CONCLUSION

Figure 5-d shows the image segmented based on the semantics. It is generated by plotting the contours of regions and sub-regions in the description built as the result of analysis.

The scene analysis system must have a scheme of control and modeling whose structure is suited to that of the scenes to be analyzed. In this paper, we have described a system which can analyze an outdoor scene with hierarchical structure. Bottom-up and top-down control were combined in the framework of region growing. Two phases were employed in the top-down process utilizing the hierarchical structure in the scene. They were effective to reduce the search space to computationally reasonable size. We are now intending to apply the system to other tasks which also have hierarchical structure.

REFERENCES

1) T.Kanade:"Model Representation and Control Structure in Image Understanding", Proc. 5th IJCAI, 1977-8.
2) Y.Yakimovsky & J.Feldman:"A Semantics-based Decision Theory Region Analyzer", Proc. 3rd IJCAI, 1973-8.
3) R.Ohlander:"Analysis of Natural Scenes", PhD. Thesis, Carnegie-Mellon University, 1975-4.
4) Y.Ohta,T.Kanade & T.Sakai:"Color Information for Region Segmentation". (in preparation).
5) A.Rosenfeld, et.al:"Scene Labeling by Relaxation Operations", IEEE Trans. SMC-6, 1976-6.
6) T.Sakai, T.Kanade & Y.Ohta:"Model-based Interpretation of Outdoor Scene", Proc. 3rd IJCPR, 1976-11.

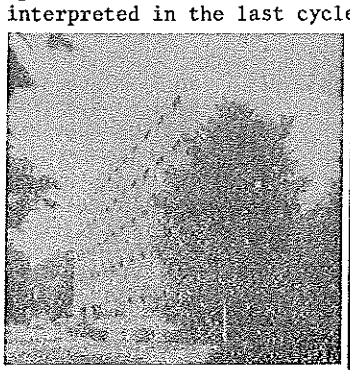
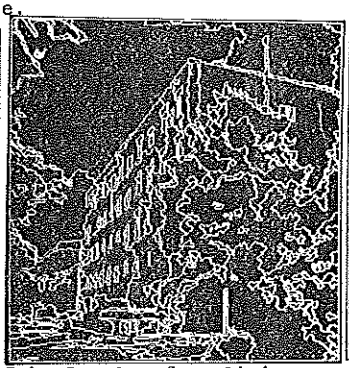
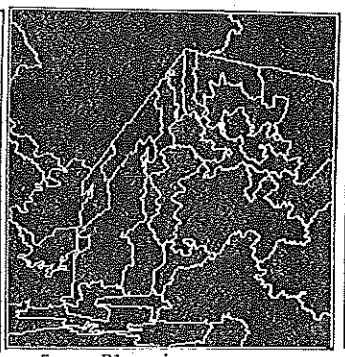


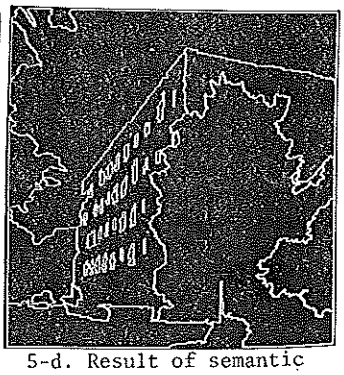
Figure 5-a. Digitized color scene.



5-b. Result of preliminary segmentation.



5-c. Plan image.



5-d. Result of semantic segmentation.