Recognition of Indoor Images Employing Qualitative Model Fitting and Supporting Relation between Objects

Keiji Yanai
Department of Computer Science,
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, JAPAN
yanai@cs.uec.ac.jp

Koichiro Deguchi
Graduate School of Information Sciences,
Tohoku University
Aramaki-aza-Aoba 01, Aoba-ku, Sendai 980-8579, JAPAN
kodeg@fractal.is.tohoku.ac.jp

Abstract

In this paper, we describe a new design of a recognition system for a single image of indoor scene including complex occlusions. In our system, first, the system estimates 3D structure of an object by fitting a 3D structure model to the image qualitatively. Next, by checking supporting relation between objects, it eliminates object candidates that are impossible to exist and estimates actual objects from their parts in the image. Then, finally, we recognize objects that are consistent with each other. We implemented the system as a multi-agent-based image understanding system. This paper describes an outline of the system and results of experiments.

1 Introduction

In usual indoor scene various objects are piling up. For example, there exists a desk on the floor and a book on the desk. Therefore, many occlusions occur, and the recognition of an indoor image must cope with them. In many conventional researches of the recognition for scene including occlusions, an exact shape model of a target object was used to recognize a single object. They fit the model to partial features and estimated total appearance of the target object. They didn't make use of spatial relation between objects.

The objective of our research is to recognize objects in a single image of real world scene including multiple objects and complex occlusions. In our research, the "recognition" means to obtain a category name of the object, such as "desk" and "chair", from real world scene. However, objects represented by a category name have many different 3D shapes. So it is impossible to prepare exact 3D models in advance. Then, we pay attention to functional structure of objects, and we use a qualitative 3D-structure model that represents essential structure for function of an object[3, 5].

But, only qualitative model fitting is not enough for scene including complex occlusions. Then, we introduce "supporting relation" that describes that which object supports which one. All objects except background objects, such as floor, wall, road, and sky, must be supported by other objects in the real world due to the gravity of the earth. We know such physical law empirically, so we can expect an existence of a desk under a workstation in the complex scene even if the desk can't be seen. We provide such physical knowledge about supporting relation with the system and make the system with an ability to estimate actual objects from parts seen in the image and eliminate object candidates that are impossible to exist.

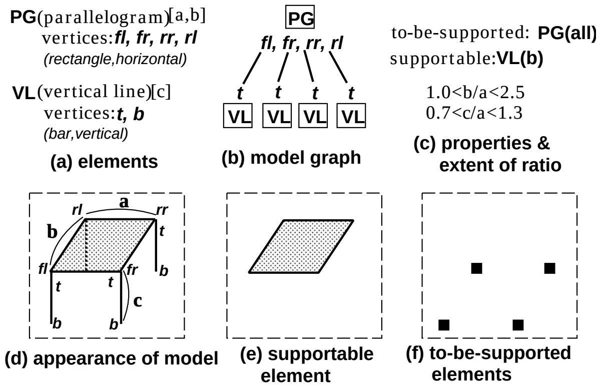
In this paper, we describe an idea to introduce such qualitative 3D model fitting and "supporting relation" checking mechanism to our multi-agent-based image understanding system, which is called MORE (Multi-agent architecture for Object REcognition)[7]. We also show results of recognition experiments.

2 Recognition strategy

2.1 Recognition by qualitative 3D model fitting

We use a prototype model that represents essential functional structure common within same kind object[5]. For example, the functional structure of a "chair" is a combination of sitting surface and one or four legs, and that of "desk" is a combination of desk face and four legs. The prototype model is represented by some model elements and a model graph. Model elements are polygons and straight line segments according to the appearance of the object (Fig.1(a)), and they have information about their real shape and their generally expected pose in the real world. A model graph represents connection relations between the model elements (Fig.1(b)). Each model has the extent of relative size among each elements and information which elements are supportable and to-be-supported (Fig.1(c)). Here, the "supportable" element and the "to-be-supported" element mean that the element can support other objects and the element must be supported by another object, respectively. These properties are used in the stage of "checking supporting relation" described later. For example, the model of "desk" has one parallelogram as its supportable element and four vertical line segments whose bottoms are its to-be-supported elements as shown in Fig.1(d)(e)(f).

For estimating regions of an object candidate, first, we extract line segments and regions by conventional methods, for example, Canny edge detector, Hough transformation, region growing segmentation method, snake and so on. Next, we search groups of line segments and regions corresponding to each element of a model. We fit the model to the group of line segments and regions extracted from the image (Fig.2). Here,



(a) elements (b) model graph (c) properties & extent of ratio

(d) appearance of model (e) supportable element (f) to-be-supported elements

Figure 1. Model representation of "desk".

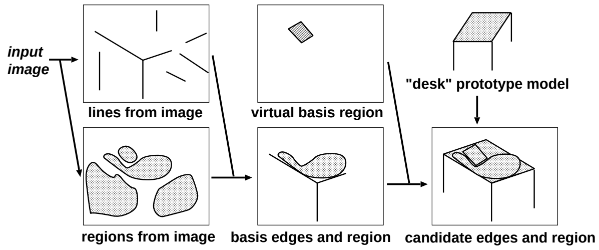


Figure 2. Estimating a "desk" candidate.

we call the lines and regions used in model fitting as **"basis edges and regions"**, and the regions where an object is estimated to exist as **"candidate region"**. A candidate region is the total region expected without occlusions. In addition, by using information about "supportable" and "to-be-supported" elements, we estimate **"supportable region"** and **"to-be-supported region"** in the image.

We compute confidence value of a candidate as a weighted sum of the ratio of basis region to candidate region of each element:

$$V_s = \sum_{i=1}^{n} W_i \frac{b_i}{e_i} \qquad (1)$$

where n is the number of elements, b_i is the number of pixels of a basis region or edge, and e_i is the number of pixels of candidate region or edge. W_i is the weighting factor, and it is provided as a priori information with each model. For example, "desk" has two elements, desk face and legs, and W_i for them are set as 0.7 and 0.3, respectively. Confidence value of a candidate is used for resolving conflict among candidates.

2.2 Checking of supporting relation

Every time the system generates a new object candidate, it examines if the "supporting relation" holds between already generated candidate and the new one. By checking supporting relation between objects, the system eliminates object candidates that are impossible to exist and estimates actual objects from parts seen in the image.

"Supporting relation" holds when the object can be considered to locate on another object and to be supported by it. Checking the "supporting relation" is carried out by examining whether the to-be-supported regions of the object is almost included in the supportable regions of another object. If so, the former object is regarded to be supported by the latter one. We present an example including a relation that "`desk is supported by floor`" in Fig.3.

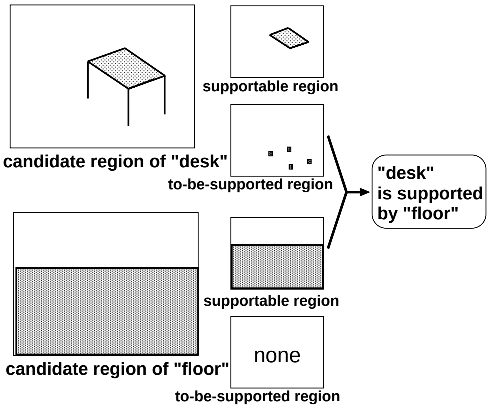


Figure 3. Checking "supporting relation".

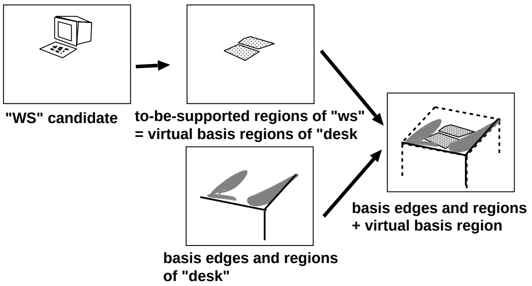


Figure 4. Estimating a "desk" candidate that supports the "workstation" candidate.

If a candidate has no supporting relation, to-be-supported regions of the candidate are regarded as **"virtual basis regions"** and the system searches a new candidate with supportable regions including the virtual basis regions(Fig.2). In short, "virtual basis regions" are regions that can be regarded as regions of supportable elements of a new candidate. Then, the system can detect a new candidate that couldn't be detected before. For example, when the system generates a "workstation (WS)" candidate with no supporting relation, it regards the to-be-supported region of "WS" as the virtual basis regions of a desk face element (Fig.4). By this mechanism, the system recognizes an object occluded by another object.

If candidates except background objects have no supporting relation, finally, the candidates are canceled.

2.3 Relational knowledge

The system has "relational knowledges". They are descriptions about relative relation generally expected between two objects. It is used for computing confidence value of relation and expecting the region where own target object exists with high possibility. It is represented by combination of "relation name", "source

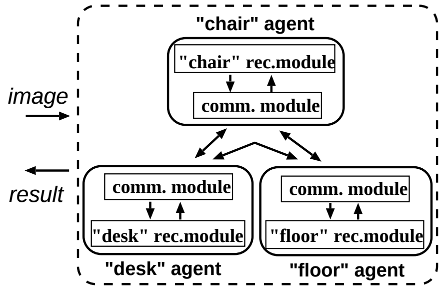


Figure 5. Basic structure of the system.

object's name" and "destination object's name". For example, ``on(book,desk)'' means "a book is usually on a desk". At present we have three types of relations, "on", "in_front_of" and "on_same_plane". The system judges whether each relation is holding by the information of supporting relations and the relative location of objects in the image.

Confidence value of relation is weighted sum of holding relations:

$$V_r = \sum_{i=1}^{r} C_i n_i \tag{2}$$

where r is the total number of relations, n_i represents if relation i is holding, it takes a value of 0 or 1, and C_i is weighting factor.

2.4 Conflict resolution

If two or more agents generate different candidates in the same region of the image, conflict occurs. Conflicting candidates are compared by their confidence values of candidates at first. If difference between the highest value and the second highest value is more than certain threshold value, all except the highest one are canceled. Otherwise, confidence values of relation are compared, then all except the highest one are canceled. If both differences are small, a temporary decision is made by comparing sizes of their regions.

3 Overview of the system

3.1 System architecture

We implemented the system based on "MORE" architecture we proposed[7]. It is multi-agent-based architecture, and the system is constructed as an assembly of agents that recognize objects from an image separately. It enables to recognize various different kinds of objects by adding agents. One agent consists of a recognition module and a communication module.

A **recognition module (RM)** has an input image, recognizes only one kind of target object, reports its region in the input image, and generates an object candidate.

A **communication module (CM)** carries out cooperation among agents. It checks supporting relations to candidates generated by other agents and resolves conflict among the agents. Every CM has relational knowledges. Using them, it computes confidence value of relation and estimates the region where own target object exists with high possibility.

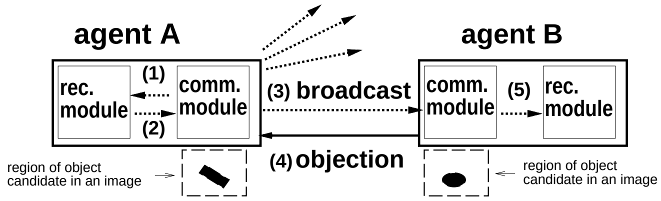


Figure 6. Flow of messages.

3.2 Processing flow

The processing flow among all the modules is message-driven. We describe the detail flow of messages in case of the example of Fig.6.

(1) Each CM sends an "initial recognition request" to each respective RM. Then, each RM starts the recognition.

(2) Every time RM finds an object candidate, it sends the candidate and its confidence value of the candidate to the CM.

(3) The CM checks if the candidate is supported by any other candidates, and broadcasts the information of the candidate for all other agents.

(4) Other agents examine if the broadcast is consistent with own object candidates. If not, the agent sends back an objection message. Then, a conflict resolution is processed between the CMs concerned.

(5) If the broadcasted object candidate has no supporting relation and the receiving agent has relational knowledge that the candidate is usually on its own object, the CM of the receiving agent sends a "conditional recognition request" to its RM. The RM starts to find object candidates with supportable regions including the virtual basis regions of the candidate.

If all modules of all the agents are in the state of waiting for a message and there is no message on communication lines, the whole recognition of the system completes. The details of this architecture were written in [7].

4 Implementation and experiments

We have implemented an experimental system with 8 agents ("desk", "chair", "wall", "floor", "book", "cup", "pen", and "work station (WS)") on PC cluster system that consists of 8 PCs(Intel Celeron 450MHz) using the PVM library[2]. In this system, each agent is implemented on each one PC.

In the experiment for the sample image no.1 (Fig.7,320x240), five objects (a "floor", a "desk", two "workstation(WS)", and a "wall") were recognized. First, two "WS" candidates were found, and the WS agent broadcasted the candidates without "supporting relation" to all other agents. Then, the desk agent, receiving them, estimated desk face by integrating basis edges and regions of "desk" (Fig.9) and to-be-supported regions of the two "WS", that is virtual basis regions of "desk" (Fig.10). It estimated candidate region by model fitting. In this way, although only part of the desk face could be seen, a "desk" had been recognized using to-be-supported regions of the objects

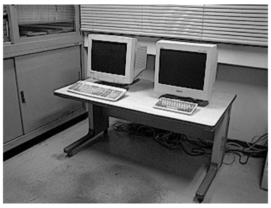


Figure 7. Sample image no.1.

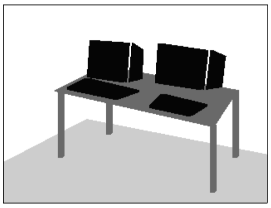


Figure 8. Recognition result.

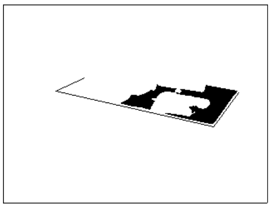


Figure 9. Basis edges and region of a "desk".

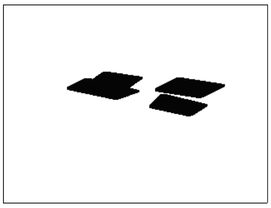


Figure 10. To-be-supported regions of two "WS".

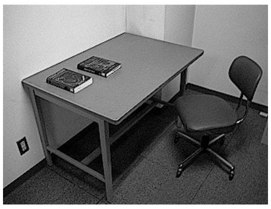


Figure 11. Sample image no.2.

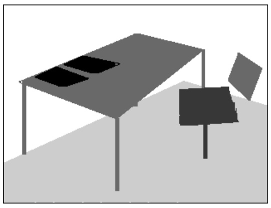


Figure 12. Recognition result.

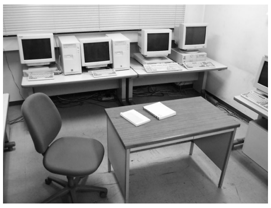


Figure 13. Sample image no.3.

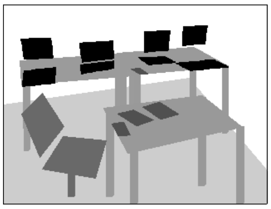


Figure 14. Recognition result.

on the "desk". In addition, in this experiment, finally, some false candidates were canceled by checking "supporting relations".

In the experiment for the sample image no.2 (Fig.11,320x240), six objects (two "books", a "desk", a "chair", a "floor", and a "wall") were recognized (Fig.12). It was difficult to recognize front and back right legs of "desk" in this image, but the system assumed that there were four legs under desk face and they were on the "floor" by the model fitting. No conflicts with other object candidates occurred for this recognition, so the assumption was regarded as true. Similarly, a "chair" was recognized on the "floor".

Sample image no.3 (Fig.13,640x480) is more complex scene, so we used a higher-resolution image than the image no.1 and no.2. Three "desks", four "WSs" and four "books" (two ones were false) were recognized (Fig.14). Especially, though almost of all the desk faces were covered with four WSs, the two back desks could be detected by using supporting relation mechanism.

5 Related work

The objective of our work is the scene recognition when exact models of target objects are not available in advance. Tenenbaum's work[6], in which segmented regions were labeled by the relaxation method, had similar objective to our work. But their work used too simple methods, and it was not available for complex images. After that, knowledge-based recognition systems, for example, the Schema System[1] and SIGMA[4], appeared. They used both models for single objects and relational knowledges among objects, and achieved an integration of bottom-up and top-down processings. Our work is similar to theirs, but their target was not indoor image but outdoor images or aerial images that scarcely include occlusions.

6 Conclusion

In this paper, we proposed a system that estimates 3D structure of a target object by fitting qualitative model qualitatively, verifies object candidates by checking "supporting relation" using "supportable regions" and "to-be-supported regions". It totally realizes a flexible recognition for real world images including complex occlusions. We have implemented the system as a multi-agent-based image understanding system on a PC cluster system.

Acknowledgments
A part of this work was supported by a grant from the Inamori Foundation.

References

[1] B. Draper, R. Collins, J. Brolio, A. Hanson, and E. Riseman. The schema system. *International Journal of Computer Vision*, 3(2):209–250, 1989.

[2] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam. *PVM: Parallel Virtual Machine.* The MIT Press, 1994.

[3] D. Kim and R. Nevatia. Recognition and localization of generic objects for indoor navigation using functionality. *Image and Vision Computing*, 16:729–743, 1998.

[4] T. Matsuyama and V. S. Hwang. *SIGMA: A knowledge-based aerial image understanding system.* Plenum Press, New York, 1990.

[5] L. Stark. Functionality in object recognition. *Computer Vision and Image Understanding*, 62(2):145–146, 1995.

[6] J. M. Tenenbaum and H. G. Barrow. Experiments in interpretation guided segmentation. *Artificial Intelligence*, 8:241–274, 1977.

[7] K. Yanai and K. Deguchi. An architecture of object recognition system for various images based on multi-agent. In *14th International Conference of Pattern Recognition*, volume 1, pages 278–281, 1998.