

A LEARNING VISION SYSTEM FOR 2D OBJECT RECOGNITION

Fumiaki Tomita
Electrotechnical Laboratory
1-1-4 Umezono, Sakura-mura, Niihari-gun
Ibaraki 305, Japan

ABSTRACT

The IMARS (Interactive Modeling and Automatic Recognition System) has been developed to recognize 2D objects in many kinds of images. Functionally necessary image processing programs are moduled in the library. Interactively testing the effectiveness of each program, the user can teach the system what objects should be recognized, how they are extracted from the image, what features should be measured for classifying them, and what feature values they have. The system stores what it learned in the model. Once the model is completed, the system can automatically analyze the same kind of image and recognize the expected objects in a top-down direction driven by the model. Since a model can be built for each kind of image, the system can analyze different kinds of images simply by changing the model.

I. Introduction

A number of image processing algorithms have been proposed to recognize specific objects in medical, aerial, or other types of images. The image recognition process is divided into two phases: the segmentation phase and the classification phase. In the segmentation phase, an image is segmented into regions, lines, and vertices. These are called image elements. The image elements correspond to the objects to be recognized. In other words, any object in an image appears as a region, a line, or a vertex. In the classification phase, each image element is classified into object category based on its features. This recognition process is thought to be highly dependent on the images analyzed. A new image processing program has to be written when a new kind of image is analyzed. It is possible, however, to avoid such programming waste by separating the image processing algorithms from image dependent information.

An image processing algorithm is the function of some parameters, and the value of those parameters depends on the images. If functionally necessary image processing algorithms (or programs) are provided, any kind of image can be analyzed simply by selecting a set of appropriate programs and activating them with the appropriate parameters. Based on this idea, the IMARS (Interactive Modelling and Automatic Recognition System) has been developed to recognize 2D objects

in many kinds of images. Interactively testing the effectiveness of each program, the user can teach the system what objects should be recognized, how they are extracted from the image, what features should be measured for classifying them, and what feature values they have. The system stores what it learned in the model. Once the model is completed, the system can automatically analyze the same kind of image and recognize the expected objects in a top-down direction driven by the model. Since a model can be built for each kind of image, the system can analyze different kinds of images simply by changing the model. IMARS can be used even when the user does not know what model can analyze his data, since a model is built in a bottom-up direction. The user learns about the objects with the system while he interactively constructs the model.

II. Overview of IMARS

IMARS is composed of the image processing program library, the image processor, the image descriptor, the model, and the interface (Fig.1). The input images may be either mono or multi spectrum images of any size and of any level within the range of computer facility. The design concept of each component is as follows:

(1) In the image processing program library, functionally necessary programs are moduled. A program specified by a user or a model is activated with the specified parameters.

(2) The image processor consists of six image memories which are used for an original image, a mask, a work area, etc. by each program.

(3) In the image descriptor, the image segmentation methods to extract the image elements and the features of the image elements are recorded. Both methods and features are referred when an element is registered to a model or when an element is matched to objects in a model.

(4) Object names, the image segmentation methods to extract the objects and the features of the objects are stored in the model.

(5) Through the interface, the original image, the image segmentation results, and the identification results are displayed. The user can communicate with the system by means of the graphics display. There are two modes: the interactive mode and the automatic mode. In the interactive mode, the user can interactively

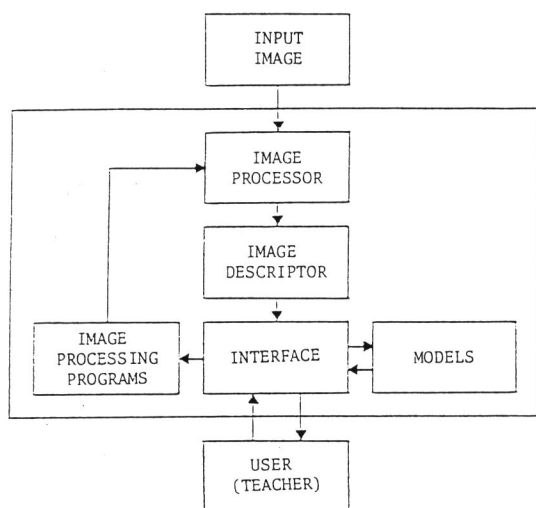


Fig. 1 Configuration of IMARS

constructs a recognition model. In the automatic mode, the system automatically analyzes an image driven by the model.

III. Model

The recognition model is represented by a graph. For example, the model which analyzes head CT images is shown in Fig.2. A node corresponds to an object and has slots for the object name, the segmentation method, the maximum and the minimum values of the parameters, and the maximum and minimum values of the features. The features with critical values for identification of the object will have a narrow range of values. Whereas, less important features will have a wider range of values. A solid link represents a parent-and-child relationship. A child object is extracted by segmenting (S) or merging (M) the parent objects. A dotted link represents an OR relationship and at least one of the linked objects is extracted by segmenting the parent object. For example, B(1)-B(6) correspond to different brain cross sections and one of them is extracted from HEAD (head CT images). A white node signifies that the object always exists and a

black node signifies that the object may not exist and is called a duty-free object.

IV. Image Segmentation

Image segmentation programs are used to extract the elements corresponding to objects. The region segmentation methods are functionally classified into physical segmentation, psychological segmentation, and semantic segmentation.

(1) Physical segmentation--First, an image is segmented into regions of different picture properties. There are the edge detection method (Tomita, 1981a) and the thresholding method (Ohlander, Price, and Reddy, 1979).

(2) Psychological segmentation--When the shape of a region is complex and difficult to describe, it can be decomposed into subregions of simple shape based on the skeleton and its cross sections (Tomita, 1981b).

(3) Semantic segmentation--When there are adjacent regions which are regarded as part of the same object, they are merged into one region. When some spatially clustering regions form an object, they are grouped into one region. When different objects are extracted as one region because they have the same picture properties and overlap, they are segmented into respective regions based on the shape of the objects.

Lines and vertices are extracted from the boundaries of regions. Lines and vertices are usually regarded as components of a region and are used to describe the detailed shape of the region. For example, a stomach (region) has components like a daiwan (line), a ikaku (vertex), and a yumon (vertex), as shown in Fig.3. The k-curvature method is used to segment the boundary of a region into lines and vertices (Rosenfeld and Kak, 1976).

Each method has its own parameters. For example, the edge detection method has parameters of a local picture property based on which difference the region is segmented, a gradient operator size, and a threshold for edge values. Almost all objects in an image can be extracted by combining those segmentation methods and adjusting the parameters.

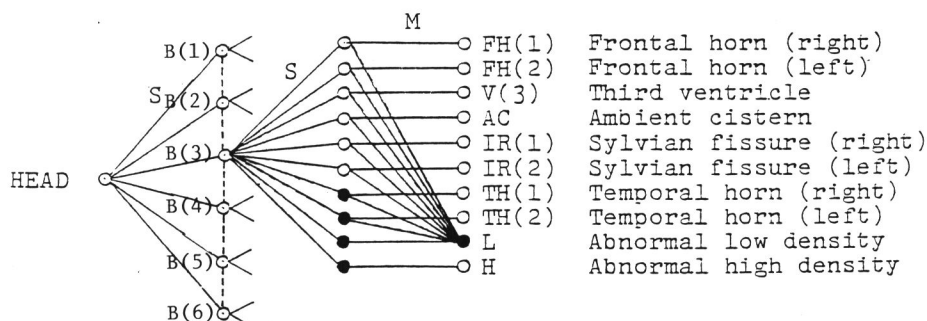


Fig. 2 Recognition model for head CT images



Fig. 3 Stomach

V. Feature Measurement

The features of extracted elements are measured for classification. There are two kinds of features: first order features and second order features. The first order features are measured for each element. They describe picture properties (gray, color, or statistical texture features), simple shape, position, and direction. The second order features are measured for a pair of elements. They describe relative values of the first order features, adjacency, symmetry, and periodicity (for structural texture features). The first order features are measured for all elements to roughly classify them. On the other hand, the second order features are measured when there is an object which cannot be identified based only on the first order features.

VI. Interactive Modeling

In the interactive mode, the user can teach the system what objects he wants to recognize, how they are extracted, what features should be measured, and what feature values they have, using training images. This procedure is called modeling and the system stores what it learned in the model.

First, the user gives the name of the image to be analyzed. The name and the picture properties of the image are stored at the root node in the model. They are used to roughly classify the image before it is analyzed in detail. (When different kinds of images are analyzed by one model, multiple root nodes are used.) Next, the user segments the image into image elements, selecting one of the image segmentation programs. The parameters are adjusted by trial-and-error until the desired elements are extracted. If there is an element corresponding to the object to be recognized, the user can incorporate it in the model by indicating the element and giving the object name. The segmentation method and the parameters are stored at the parent object node in the model. The new node corresponding to the indicated object is linked to the parent object node and the name and first order features are stored there. The above procedure is repeated until all necessary objects are registered in the model.

After the initial model is built, the user can test and improve the model by having the system automatically analyze succeeding images using each level of the model. When the system fails to recognize an expected object, it demands the user's help. The user can take one of the following actions to improve the model:

(1) Renovation of parameters--When an expected object was not extracted well, segmentation is retried by interactively adjusting the parameters and the maximum or minimum values of the parameters in the model is revised.

(2) Renovation of first order features--When an expected object was extracted but was not identified, the user must revise the maximum or minimum values of the first order features by registering the unidentified element.

(3) Renovation of second order features--When an expected relationship was not found for a pair of objects, the user revises the maximum or minimum values of the second order features by indicating the corresponding pair.

(4) Registration of OR relationship--When there are objects, at least one of which is extracted by segmenting the parent object, the user must inform the system that they have an OR relationship. The multiple root nodes (images) have OR relationships.

(5) Registration of duty-free object--When there is an object which does not always exist, the user must inform the system that it is a duty-free object.

When there is an identification error, the user must take one of the following actions:

(1) Addition of constraints--If there is a second order feature which can distinguish two confusing objects, the user must teach the system to measure the feature for either or both objects.

(2) Rebuilding the model--Otherwise, it is necessary to change the structure of the model.

Modeling is repeated until all the training images are analyzed properly by the model. A flexible model is built by making both the parameters of the segmentation programs and the features of the objects have a range of values and by making the objects have OR relationships.

VII. Automatic Recognition

Once the recognition model is completed, the system can automatically analyze the same kind of image driven by the model. Since the parameters used in segmentation have a set range, the system must select the best value for each parameter within the range so that it could extract the expected objects. Let a parameter have the maximum value P_{max} and the minimum value P_{min} . The system tries P_{max} , $(P_{max}+P_{min})/2$, or P_{min} for the parameter. If there are n parameters, it tries a maximum of 3^n cases until it finds all expected objects. When it fails to find an expected object by adjusting the parameters, it backtracks. When it fails to find an expected object by backtracking, it gives the message:

No model matches !

In this case, the user must teach the system how to recognize the object and improve the model in the interactive mode.

VIII. Experiments

We first used IMARS to recognize the brain parts in head CT images and to detect abnormalities. Since the shape and the position of the brain parts differ depending on individuals and ages, a flexible model is necessary to recognize them. The recognition model for head CT images is shown in Fig.2.

The edge detection method was used to segment

head CT images into regions of different picture properties. Since there is a high contrast between the skull and the brain, the image in Fig.4(a) was first segmented using a 3x3 gradient operator and a region corresponding to the brain was extracted, as shown in Fig.4(b). The features of the extracted regions are measured to be matched with those of the objects in the model. As a result, the brain B(3) is identified among the regions, as shown in Fig.5(a). Since there is a texture in the brain, the brain is next segmented using a 7x7 operator and the regions corresponding to the brain parts are extracted, as shown in Fig.4(c). The candidates for cerebrospinal fluid parts (frontal horns, ventricles, ambient cistern, silvian fissures, etc.) and an abnormal low density part are identified among the regions, as shown in Fig.5(b). Here, symmetry (a second order feature) is used to distinguish between the normal cerebrospinal fluid parts and the abnormal part because the normal cerebrospinal fluid parts have a symmetrical axis about the vertical center line of the brain but abnormal parts usually do not. If there are adjacent regions which are identified as the parts of the same objects, they are merged. Finally, the cerebrospinal fluid parts and an abnormal low density part (smeared black) are recognized, as shown in Fig.5(c).

IX. Conclusion

It is assumed that functionally necessary programs are provided in the library. Even if a necessary program was lacked, it can be easily found since the effectiveness of each existing program is interactively tested. IMARS does what it was taught by the user but it has no ability of inference yet. Although the current IMARS was designed to recognize 2D objects, the system will be extended to recognize 3D objects.

REFERENCES

- [1] Tomita, F., "Region extraction by the edge detection method," *Proc. the 23th Joint Convention of the IPSJ*, 1981, pp.723-724 (in Japanese).
- [2] Ohlander, R., Price, K., and Reddy, O., "Picture segmentation using a recursive region splitting method," *Computer Graphics and Image Processing*, 8:3, (1978) 313-333.
- [3] Tomita, F., "Hierarchical description of textures," *Proc. IJCAI-81*, Canada, August, 1981, pp.728-733.
- [4] Rosenfeld, A. and Kak, A. C., *Digital Picture Processing*, New York: Academic, (1976), 371-379.

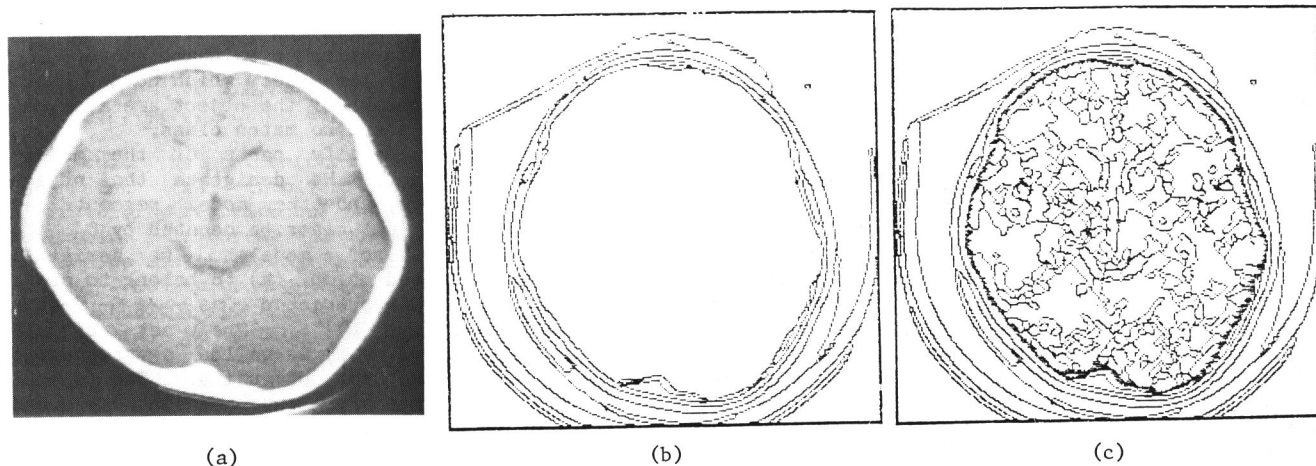


Fig. 4 Image segmentation

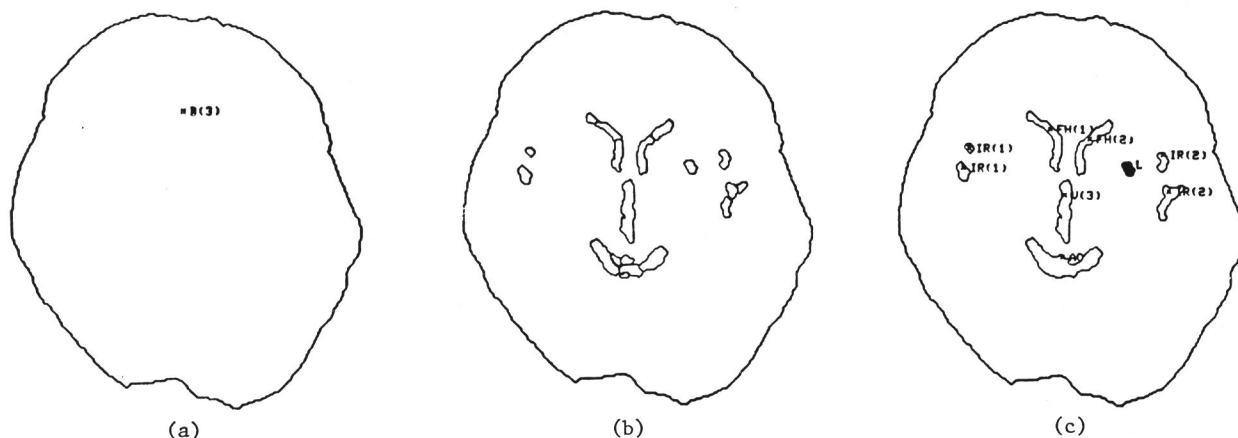


Fig. 5 Object recognition