We have developed an expert system that can automatically construct image-processing algorithms by combining elemental processing modules included in an image-processing software package. This system has two kinds of knowledge databases: one contains production rules for generating a rough flow of image processing based on experts' experience; and the other is a set of frames representing the function of each of the package's image processing modules.

Constructing an image-processing expert system involves the following practical problems:

(1) Image characteristics
We must develop an easy-to-use interface that lets the system can understand image characteristics through questions and answers.

(2) Processing flow inference

We must put the expert's knowledge into a database and construct an inference mechanism able to produce the processing flow using the knowledge database.

(3) Processing module selection

We must put knowledge about each processing module's functions into a database and construct an inference mechanism able to select the most appropriate modules using the knowledge database.

In our system, image characteristics are communicated in interactive communication. For flow inferring, the expert's experience resides in a knowledge database as production rules. For module selection, knowledge about module functions resides in another knowledge database as a frame structure. The inference mechanism for both processing flow inference and processing module selection is achieved in the form of a production system.

The system asks the user about image characteristics. Then the system grasps image characteristics based on the answers to those questions. The image characteristics are expressed in the attribute name and the attribute value in the image attribute description section.

The image-processing goal that the user presents is replaced by a combination of image-processing modules through repeated processing flow inference and processing module selection. The goal is what the user hopes to accomplish by image processing. In processing flow inference, goal
decomposition -- replacing an abstract goal with a combination of less abstract goals -- is repeated recursively until all goals become definite enough to be directly replaced by concrete modules. An abstract goal presented by the user is decomposed into a combination of other goals, which, in turn, are decomposed. There is usually more than one way to decompose a goal. The optimum goal combination is selected using knowledge based on the expert's experience.

The decomposition sequence must follow the actual image-processing flow, because image characteristics change during image processing, and subsequent goal decomposition must be performed based on the altered image characteristics.

Once a goal that can be replaced directly by a concrete module is derived, processing module selection step is activated. The system selects the optimum module for achieving the goal, derived from goal decomposition, by using knowledge about the module functions.

Because the actual question and answer step is the decisive aspect of the processing flow inference section and the processing module selection section, question-and-answer content is not given in a prototyped static format, but changes with the result of inferring. The image attribute description section records image characteristics and the history of the ongoing inferring.

The processing flow inference section makes inferences by collating image characteristics and contents in the knowledge database. This is controlled by a stack where goals to be decomposed are stored in a definite sequence. When the user enters a processing goal, the system pushes the goal onto the stack, which is initialized as empty. The system then pops the
first goal from the stack and selects the corresponding rules. If there are no rules, the goal is posted to the processing module selection section, which is then activated. When there are rules in the knowledge database, the system collates each rule's condition field contents and image characteristics in the image attribute description section and, when they match, selects this rule. Goals indicated in the decomposition indication field are then pushed onto the stack in reverse order so that the inferring order agrees with the order of actual image processing. The goal is again popped from the stack, and the process is repeated.

The knowledge database on modules was created in the hierarchical frame structure. A generic frame is assigned for each goal that can be replaced by a concrete module and an individual frame assigned for each actual module used to achieve the goal.

The generic frame consists of two slots: the module name indication slot where related module names are stored and the image attribute name slot where image attribute names referenced in selecting the optimum modules are stored. Each individual frame has slots corresponding to the image attribute names in a generic frame. The image attribute slot gives values of image attributes as conditions under which the module can be used, together with values for image attributes resulting from module use.

The module is actually selected by collating the frame content and image characteristics in the image attribute description section. The generic frame corresponding to the goal posted from the processing flow inference section is selected, then the system collates attribute values recorded in the image attribute description section and in each individual
frame, when they match, the system selects the relevant module and replaces values in the image attribute description section based on individual frame contents.

The knowledge database of the expert's experience consists of 260 production rules, while that on module functions consists of 160 frames. The system's ease of use promises to expand its techniques a wide range of applications, and is expected to greatly improve productivity in constructing algorithm.