PERGAMON

Pattern Recognition 35 (2002) 1793–1806

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Scenery image recognition and interpretation using fuzzy inference neural networks

Hitoshi Iyatomi, Masafumi Hagiwara*

Department of Information and Computer Science, Keio University 3-14-1 Hiyoshi, Yokohama 223-8522, Japan

Received 5 July 2001

Abstract

In this paper, we propose a new image recognition and interpretation system. The proposed system is composed of three processes: (1) regional segmentation process; (2) image recognition process; and (3) image interpretation process. As a pre-processing in the regional segmentation process, an input image is divided into some proper regions using techniques based on K-means algorithm. In both the image recognition and the interpretation processes, fuzzy inference neural networks (FINNs) working in parallel are employed to achieve a high level of recognition and interpretation. Scenery images are used and it is confirmed that the system has an average of 71.9% accuracy rate in the recognition process and good results in the interpretation process without heuristic knowledge. In addition, it is also confirmed that the proposed system has an ability to extract proper rules for the image recognition and interpretation. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Neural network; Fuzzy inference; Scenery image; Image recognition; Image understanding

1. Introduction

Image understanding is very important because it has a wide range of applications such as robotics, human interface, and multimedia. A lot of studies have been made. Since many conventional studies use pattern matching techniques, however, they require heuristic knowledge in advance [1–6]. As a result, these systems might show superior recognition results only in limited situations.

Based on such problems, several objects recognition methods employing learning ability of neural networks have been proposed [7–11]. Peng et al.'s system [7] has a hierarchical structure from a segmentation process to a pattern matching one. This system determines segmentation parameters by learning and extracts car areas from images with complicated background. However, pattern matching is carried out in the recognition process: the system is rule-based.

Different from these studies, Ref. [8] does not require domain knowledge in advance and has shown fairly good recognition results. Since it uses a hierarchical neural network with back propagation learning algorithm, the roles of hidden units are unclear. Therefore, extraction of appropriate rules or knowledge is difficult.

Automatic extraction of rules or knowledge is also very important although it is difficult. Several studies have been reported. One of the promising approaches is the combination of learning ability of neural networks and rule processing ability of fuzzy logic theory [12–16]. Among them, fuzzy inference neural network (FINN) [15] which can extract fuzzy rules automatically has a simple structure and superior performance. We made a preliminary study on scenery image recognition and extraction of rules using FINNs [17]. Since the recognition and rule extraction in the system are based on pixel

* Corresponding author. Tel.: +81-45-566-1762; fax: +81-45-566-1747.
E-mail address: hagiwara@soft.ics.keio.ac.jp (M. Hagiwara).

PII: S0031-3203(01)00171-6

by pixel, shape information cannot be treated. Therefore, obtained rules are very simple and the system requires a huge amount of computations.

In this paper, we propose a sophisticated recognition and rule extraction system from scenery images using FINNs. The proposed system can treat shape information because of the regional segmentation as a pre-processing. In addition, high level image recognition and interpretation are possible and the system has ability of additional learning.

Regional segmentation subject is itself a profound one and a lot of studies have been carried out [18–25]. The image retrieval system [26] uses K-means algorithm based on I_1, I_2, I_3 color set and achieved good segmentation results. We employ the improved method [20] of K-means algorithm [18] using I_1, I_2, I_3 color set [26] that can handle many characteristic values at once and has low dependence on the parameters.

This paper is organized as follows. In Section 2, we explain the learning system and the FINNs. An overview of the proposed system is provided in Section 3. In Section 4, the performance is shown and discussed. Concluding remarks are provided in Section 5.

2. Learning system

Since most of the image recognition systems use pre-implemented rules, they might have excellent performance in limited situations. One way to obtain high generalization ability for the systems is by inputting more and more rules to the systems in advance. However, the number of rules in such rule-based systems might be huge and it is almost impossible to describe rules to cover any situation.

Fig. 1 shows the comparison of a rule-based system and a learning-based system. Generally, the learning-based systems are superior to the rule-based ones because the learning-based systems can construct and modify effective rules.

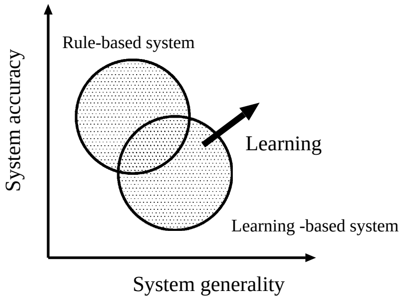
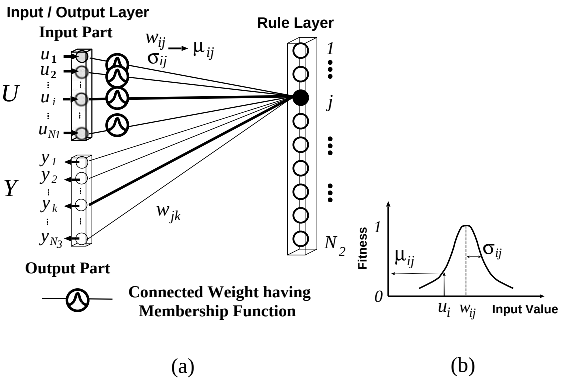


Fig. 1. Rule-based and learning-based systems.



(a) (b)

Fig. 2. Structure of FINN and the membership function.

The proposed system aims at high level image recognition and interpretation with FINN [15] based networks which can extract and use rules automatically.

2.1. Fuzzy inference neural network (FINN)

The proposed system consists of several modified FINNs [15] which can divide input–output data space by self-organizing learning and extract fuzzy rules automatically.

Fig. 2(a) shows the structure of FINN. It consists of two layers. One is the input–output (I/O) layer and another is the rule-layer. The I/O layer consists of the input- and the output-part. Each node in the rule-layer represents one fuzzy rule. Weights from the input-part to the rule-layer and those from the rule-layer to the output-part are fully connected and they store fuzzy if-then rules.

Membership functions as premise parts are expressed in the weights from the input-part to the rule-layer. Each weight from the rule-layer to the output-part corresponds to the estimated value of each rule. In short, the weights from the input-part to the rule-layer indicate if-parts of fuzzy if-then rules and those from the rule-layer to the output-part indicate then-parts. The shapes of membership functions are adjusted automatically in the learning process explained in Section 2.1.2.

2.1.1. Behavior of FINN

Suppose that the number of neurons in the input-part, which is equal to the dimension of the input data, is N_1, the number of rules is N_2, and the number of neurons in the output-part, which is equal to the dimension of the output data, is N_3. The input data to the FINN is expressed as follows:

$$\boldsymbol{U} = (u_1, u_2, \ldots, u_i, \ldots, u_{N_1})^{\mathrm{T}}. \tag{1}$$

The subscripts i, j, and k refer to the nodes in the input-part, those in the rule-layer, and those in the output-part.

Fig. 2(b) shows an example of a membership function. The bell-shaped membership function represents the if-part of fuzzy rule, which is placed between the input node i and the node j on the rule-layer. The membership function is expressed as

$$\mu_{ij} = \exp\left(-\frac{(u_i - w_{ij})^2}{\sigma_{ij}^2}\right), \quad j = (1, 2, \ldots, N_2), \tag{2}$$

where w_{ij} is the center value of the membership function, σ_{ij} indicates the width of the membership function adjusted in the learning process explained in Section 2.1.2.

In the rule-layer, the degree of the jth rule ρ_j is calculated.

$$\rho_j = \min[\mu_{1j}, \mu_{2j}, \ldots, \mu_{ij}, \ldots, \mu_{N_1 j}]. \tag{3}$$

Then, the inference result of the kth node in the output-part, $\hat{y}_k$, is calculated by the following equation.

$$\hat{y}_k = \frac{\sum_j^{N_2}(w_{jk}\rho_j)}{\sum_j^{N_2}\rho_j}, \quad k = (1, 2, \ldots, N_3), \tag{4}$$

where w_{jk} is the weight between the jth node in the rule-layer and the kth node in the output-part. The w_{jk} corresponds to the estimated value of the jth rule for the kth node in the output-part. The logical form of the fuzzy inference if-then rules is given as

If u_1 is $\tilde{w}_{1j}$, and u_2 is $\tilde{w}_{2j}$, and $\ldots, u_i$ is $\tilde{w}_{ij}, \ldots, u_{N_1}$ is $\tilde{w}_{N_1 j}$ **then** $\hat{y}_k$ is w_{jk}, where $\tilde{w}_{ij}$ means the value near w_{ij}. It should be noted here that it depends on the value of σ_{ij}.

2.1.2. *Learning process in FINN*

There are two learning phases in FINN. The first one is the self-organizing learning phase and the other is the LMS (least mean square) learning phase.

First, in the self-organizing phase, the center values of membership functions which correspond to the if-part and the estimated values which correspond to the then-part are determined by Kohonen's algorithm [14] temporarily. Second, LMS learning phase (supervised learning phase) is executed to reduce the total mean-square error of the network to finely adjust the weights and the shapes of membership functions.

2.1.2.1. Self-organizing learning phase. In this phase, Kohonen's self-organizing algorithm is applied to roughly classify the input data. For the sake of simplicity, we define the $(N_1 + N_3)$ dimensional input vector $\boldsymbol{X}$ as follows:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{U} \\ \boldsymbol{Y} \end{bmatrix}. \tag{5}$$

Here, the vector $\boldsymbol{Y}$ is the desired data vector in the output-part of the I/O layer described as

$$\boldsymbol{Y} = (y_1, y_2, \ldots, y_k, \ldots, y_{N_3})^{\mathrm{T}}. \tag{6}$$

The jth weight vector is defined as follows:

$$\boldsymbol{W}_j = \begin{bmatrix} \boldsymbol{W}_j^{(12)} \\ \boldsymbol{W}_j^{(21)} \end{bmatrix}, \tag{7}$$

where $\boldsymbol{W}_j^{(12)}$ is the weight vector from the input-part in the I/O layer to the jth node in the rule-layer, and $\boldsymbol{W}_j^{(21)}$ is the weight vector from the jth node in the rule-layer to the output-part in the I/O layer. They are expressed as

$$\boldsymbol{W}_j^{(12)} = (w_{1j}, w_{2j}, \ldots, w_{ij}, \ldots, w_{N_1 j})^{\mathrm{T}} \tag{8}$$

and

$$\boldsymbol{W}_j^{(21)} = (w_{j1}, w_{j2}, \ldots, w_{jk}, \ldots, w_{jN_3})^{\mathrm{T}}, \tag{9}$$

respectively.

According to Kohonen's self-organizing feature map algorithm, the j^*th node in the rule-layer is regarded as the *winner* if it can suffice the following equation.

$$||\boldsymbol{W}_j^* - \boldsymbol{X}|| = \min_j(\alpha||\boldsymbol{W}_j - \boldsymbol{X}||), \tag{10}$$

where α is the parameter which adjusts the influence of inputs and $\boldsymbol{W}_j^*$ is the weight vector of the winner in the rule-layer. These weights are updated by the following equation:

$$\boldsymbol{W}_j(t+1) = \boldsymbol{W}_j(t) + \varepsilon_{self}(t)h(j, j^*, t)(\boldsymbol{X} - \boldsymbol{W}_j(t)), \tag{11}$$

where $\varepsilon_{self}(t)$ is a gradually decreasing learning function, and $h(j, j^*, t)$ is the neighborhood function which is given by

$$h(j, j^*, t) = \exp\left(-\frac{d^2(j, j^*)}{\sigma(t)^2}\right), \tag{12}$$

where $d(j, j^*)$ is the Euclidean distance between the jth node and the j^*th node in the rule-layer, $\sigma(t)$ is the width parameter which is gradually decreased. The range of $h(j, j^*, t)$ reduces gradually as the learning proceeds.

2.1.2.2. Supervised learning phase. In the supervised learning phase, by means of LMS algorithm, mean square error between outputs of the network and the desired signals is reduced to adjust the parameters such as those to determine the shape of the membership functions explained before.

The goal of the LMS learning phase is to minimize the following error function E. Here, we regard minimizing E as minimizing error function E_p, which indicates the sum of errors for each learning pattern p.

$$E=\sum_{p} E_p, \tag{13}$$

$$E_p=\frac{1}{2}\sum_{k}^{N_3}(y_k-\hat{y}_k)^2, \tag{14}$$

where y_k is the desired output at the kth node in the output-part and $\hat{y}_k$ is the output inferred by FINN.

Suppose that x is a parameter to be adjusted, the LMS learning algorithm can be expressed as

$$x(t+1)=x(t)-\varepsilon_{LMS}\frac{\partial E}{\partial x}, \tag{15}$$

where ε_{LMS} is the learning constant.

According to the LMS learning principle, the estimation value of jth rule node is updated as

$$w_{jk}(t+1)=w_{jk}(t)+\varepsilon^{w}_{LMS}(y_k-\hat{y}_k)\frac{\rho_j}{\sum_n^{N_2}\rho_n}. \tag{16}$$

The center and the width of membership functions are updated as

$$\begin{aligned} w_{ij}(t+1) &= w_{ij}(t)+\varepsilon^{w}_{LMS}\sum_{k}^{N_3}(y_k-\hat{y}_k)\\ &\quad\times\left(\frac{w_{jk}\sum_n^{N_2}\rho_n-\sum_n^{N_2}(w_{nk}\rho_n)}{(\sum_n^{N_2}\rho_n^2)}\right)\\ &\quad\times q_{ij}\mu_{ij}\frac{2(u_i-w_{ij})}{\sigma_{ij}^2} \end{aligned} \tag{17}$$

and

$$\begin{aligned} \sigma_{ij}(t+1) &= \sigma_{ij}(t)+\varepsilon^{\sigma}_{LMS}\sum_{k}^{N_3}(y_k-\hat{y}_k)\\ &\quad\times\left(\frac{w_{jk}\sum_n^{N_2}\rho_n-\sum_n^{N_2}(w_{nk}\rho_n)}{(\sum_n^{N_2}\rho_n^2)}\right)\\ &\quad\times q_{ij}\mu_{ij}\frac{2(u_i-w_{ij})^2}{\sigma_{ij}^3}, \end{aligned} \tag{18}$$

respectively, where

$$q_{ij}=\begin{cases} 1 & \text{for } \rho_j(=\min[\mu_{1j},\mu_{2j},\ldots,\mu_{N_1j}])=\mu_{ij},\\ 0 & \text{elsewhere.}\end{cases} \tag{19}$$

3. Proposed system

Fig. 3 shows the overview of the proposed system. It consists of three processes:

1. Regional segmentation process.
2. Image recognition process.
3. Image interpretation process.

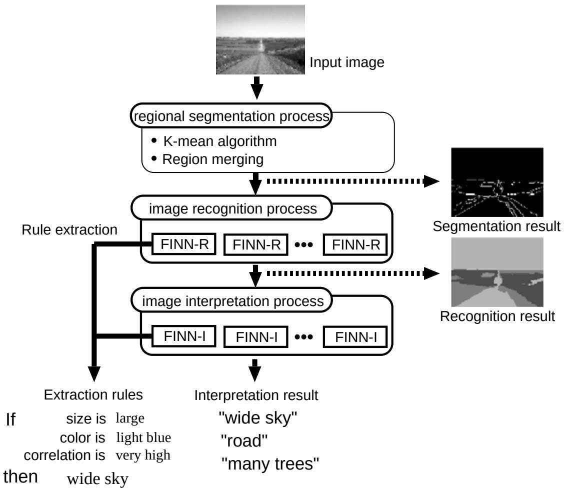


Fig. 3. Overview of the proposed system.

In the regional segmentation process, an input image is divided into some regions, which greatly reduces the amount of computations. In the image recognition process, fuzzy inference is carried out in every region using plural FINNs working in parallel. In the last process, the system outputs the interpretation results of the image using recognition results from FINNs. In addition, rules related to the scenery image recognition and interpretation are extracted.

3.1. Regional segmentation process

In the proposed system, since the processing is performed region by region, segmentation is carried out as a pre-processing. In this process, an input image is divided into a number of proper regions through the following four stages:

1. Regional division by K-means algorithm.
2. Regional merging using majority filter.
3. Small region merging.
4. Regional merging by edge comparison.

This region segmentation is carried out in a bottom–up way: dividing small regions first and then they are combined to form proper regions.

3.1.1. Region division by K-means algorithm

The proposed system uses five characteristics for K-means algorithm: three color references, I_1,I_2 and I_3 and two positional references, x and y.

The color set of I_1,I_2 and I_3 is more independent of each other than that of RGB, each of which has an intensity (V) information. Therefore, it is suggested in Ref. [1] that this color set is suitable to deal with

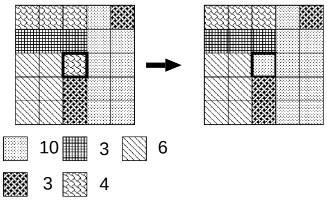


Fig. 4. Majority filter (5 × 5).

textured real images, human image, etc. and the good segmentation results have been demonstrated. In addition, the scenery image retrieval system [26] uses K-means algorithm with I_1, I_2 and I_3 color set. It also achieved very reasonable results.

The color references I_1, I_2 and I_3 are calculated as follows:

$$I_1 = (R + G + B)/3,$$

$$I_2 = R - B,$$

$$I_3 = (2G - R - B)/2. \tag{20}$$

Then, the characteristic distance L between a pixel and the cluster by K-means algorithm is calculated as follows:

$$L^2 = K_l\{(I_1 - \bar{I}_1)^2 + (I_2 - \bar{I}_2)^2 + (I_3 - \bar{I}_3)^2\} \\ + K_p\{(x - \bar{x})^2 + (y - \bar{y})^2\}, \tag{21}$$

where $\bar{x}$ and $\bar{y}$ express the mean position of the cluster and $\bar{I}_1, \bar{I}_2$ and $\bar{I}_3$ are the mean color components. K_l and K_p are the weight value for the position and that for the color characteristics, respectively.

3.1.2. Regional merging using majority filter

It is pointed out that one of the disadvantages of K-means algorithm is that it produces a lot of tiny regions around big ones [20]. In order to reduce the phenomena, the method called "majority filter" [20] that combines those unnecessary small regions is employed in the proposed system. The concept of majority filter is shown in Fig. 4. In this figure, an attribute of the pixel located in the center of the rectangle is determined in order that it equals the attribute of pixels contained most in the rectangle.

3.1.3. Small region merging

Owing to the majority filter, the number of regions is reduced largely, however, there still remains a lot of tiny regions. In the proposed system, such tiny regions whose sizes are smaller than the pre-determined S_{th}% of the original image are combined to the largest neighboring region.

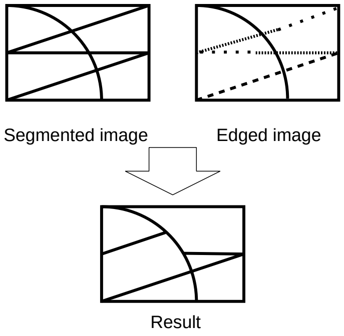


Fig. 5. Region merging by edge comparison.

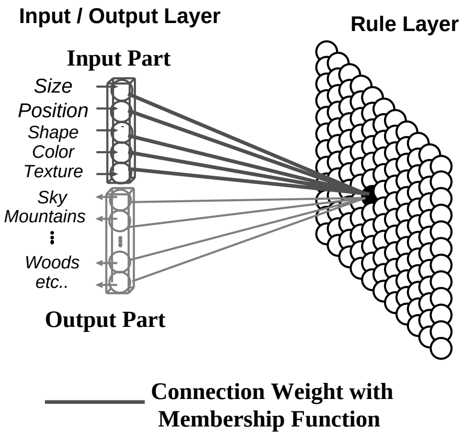


Fig. 6. Structure of FINN-R.

3.1.4. Regional merging by edge comparison

Precision of the segmentation is not enough especially for the areas including gradation such as a cloud in a sky. In this stage, the segmentation result obtained so far is compared with the edged original image.

The idea for this procedure is shown in Fig. 5. Here, we define the edge intensity threshold E_{th} and the region merging rate threshold R_{th}(%). Only when the edge whose intensity exceeds E_{th} and it overlaps R_{th}% with the boundary obtained by segmentation, these two regions are retained as they are. Otherwise these regions are combined. This process is very effective and contributes to reduce recognition error.

3.2. Image recognition process

In this process, each region in the segmented image is given the recognition label. The proposed system uses networks for image recognition based on FINN [15]. We call this network as "FINN-R" (Recognition). Fig. 6 shows the structure of FINN-R. The rule-layer in FINN-R consists of the neurons arrayed two dimensionally and both the behavior and learning algorithms are the same as the original FINN [15,17].

Table 1
Inputs of FINN-R

Size	s	
Position	x, y	
Shape	σ_x, σ_y	
Mean color	$V(Intensity), H(Hue), S(Saturation)$	
Texture	contrast	$0^\circ, 45^\circ, 90^\circ, 135^\circ$
	uniformity	$0^\circ, 90^\circ$
	correlation	$0^\circ, 90^\circ$
	entropy	$0^\circ, 45^\circ, 90^\circ, 135^\circ$

Table 2
Outputs of FINN-R (recognition labels)

Sky	Woods
Cloud	Grass
Water	Shadow
Mountains	Snow
Rocks	Light

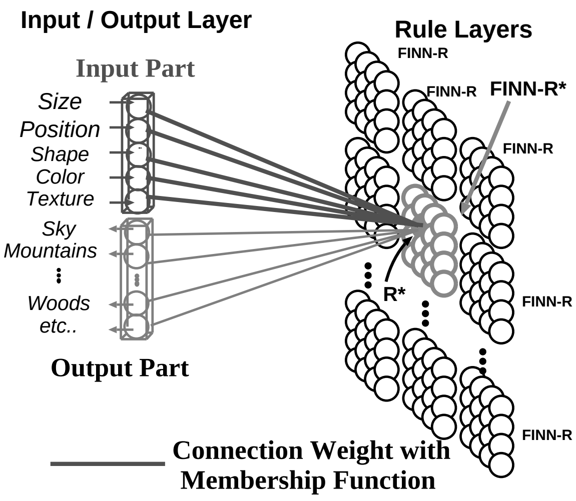


Fig. 7. Image recognition network.

3.2.1. *Inputs of FINN-R*

The inputs of FINN-R and the outputs as recognition labels are summarized in Tables 1 and 2, respectively. FINN-R, for every region, receives a total of 20-dimensional information such as the size, shape, mean color and texture characteristics, and outputs the recognition label such as sky, woods, and so on.

Each FINN adjusts the center value and width of its fuzzy membership function automatically during the LMS learning phase. The width of each membership function corresponds to the diversity of the input of the fuzzy rule. When the width of membership function is narrow, the input value is sensitive and has a large effect on the results. On the other hand, when the width is large, it means that the input is not very important. Therefore, it is possible to estimate the importance of each input.

3.2.2. *Learning scheme of FINN-R*

Since FINN-R requires learning, first, images for the learning are segmented in regional segmentation process

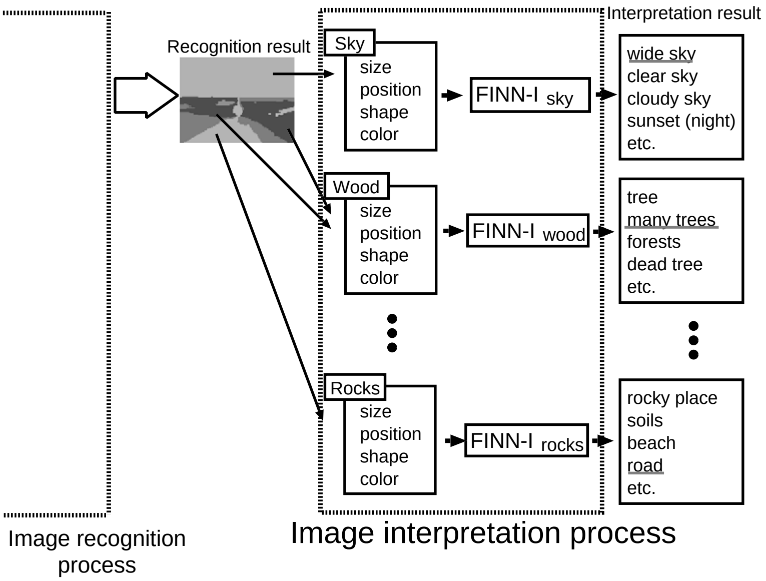


Fig. 8. An overview of the interpretation process.

(written in Section 3.1). Twenty characteristics from each region are used as an input vector $\boldsymbol{U}$ and the corresponding label, such as "sky", "mountain" and so on, is used as the teaching vector $\boldsymbol{Y}$ of FINN-R. The number of elements in $\boldsymbol{Y}$ equals that of recognition labels.

3.2.3. Usage of multiple FINN-Rs

The proposed system employs multiple FINN-Rs working in parallel because of the advantage of effectiveness in the learning, generalization and extension ability of the system.

Fig. 7 shows the image recognition network using multiple FINN-Rs. Since each FINN-R receives the same input, there is one common I/O layer. All the FINN-Rs have learnt different images at the learning process, obtained networks are also different. The system selects such an FINN-R that has the rule R^* with the highest fitness value ρ. We denote this selected FINN-R as "FINN-R*". In short, the system can use the most proper FINN-R for each region automatically.

In addition, usage of multiple FINN-Rs has other merits. One such merit is that since one FINN-R is a fundamental component of the system, it can be added or deleted easily: additional learning is possible. Another merit is that smaller networks are easy to maintain.

3.3. Image interpretation process

In this process, an input image is interpreted using results of all FINN-Is. Fig. 8 shows the overview of this process. In this process, the regional information is extracted and then it is input to the FINNs for interpretation. We call this network as FINN-I (interpretation).

Multiple FINN-Is are prepared in the same number as the recognition labels: the proposed system uses 10 FINN-Is. Here, for example, an FINN-I for the sky area interpretation is denoted as "FINN-Isky", in the same way, that for cloud area is "FINN-Icloud" and so on.

3.3.1. Inputs of FINN-I

The interpretation labels are summarized in Table 3. Thirteen characteristics such as the regional size, position, shape, and mean color are used for the inputs to FINN-Is. On the other hand, several interpretation labels including "etc." are used.

3.3.2. Learning scheme of FINN-I

Fig. 9 shows an example for learning of FINN-I. Since the "wide sky" area in Fig. 9 relates to "sky", 13 characteristics mentioned in Section 3.3.1 are used as learning vector $\boldsymbol{U}$ and "wide sky" is used as the teaching vector $\boldsymbol{Y}$ for the FINN-Isky. In the same way, the other areas are learnt in the corresponding FINN-I.

Table 3
Interpretation labels (FINN-Is output)

Recognition label	Interpretation label
Sky	Wide sky
	Clear sky
	Cloudy sky
	Sunset/night
Cloud	Sunny
	Cloud
	Heavy cloud
	Night cloud
Water	Ocean
	Lake
	River
	Water (etc.)
Mountains	Majestic mountain
	Steep mountain
	Far mountain
	Peninsula
Rocks	Rocky mountain
	Gravel/mud
	Sandy beach
	Gravel road/road
Woods	Forest
	Tree
	Many trees
	Dead/red tree
Grass	Grassy plain
	Flowers
	Dark prairie
	Field
Shadow	Dark
	Shadow
	Mountain shadow
Snow	Snowy mountain
	Covered snow
Light	Sun
	Glaring
	Dazzling

3.3.3. Getting interpretation results

Now we explain the inference process. The largest inference result of each FINN-I is considered as the most reliable in each corresponding label. Among them, the interpretation labels within top five and those of the corresponding estimated values (w_{jk}, see Eq. (4)) for the rules that are over 0.5 are used as the final interpretation.

If FINN-Isky outputs "wide sky (0.7)", FINN-Imountain outputs "majestic mountain (0.8)", and

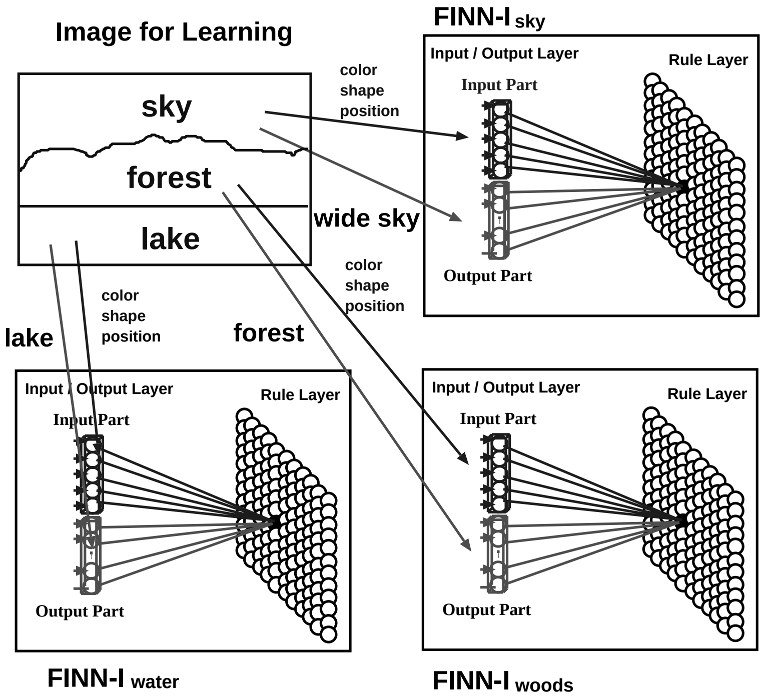


Fig. 9. Learning of FINN-I.

FINN-Iwater outputs "lake (0.4)" respectively, for example, then the final interpretation result is given as "wide sky" and "majestic mountain".

3.4. Rule extraction concerning image recognition and interpretation

The proposed system can extract fuzzy if-then rules automatically from the networks by analyzing the inputs and outputs of FINN-Rs and FINN-Is.

We explain the rule extraction process using Fig. 10. In the image recognition process, as we explained before, information on each segmented region is input to all the FINN-Rs and then the winner network FINN-R* which has the rule R^* with the largest fitness value is selected. In this case, the largest rule R^* is "if size is very big, color is blue and correlation is high, then (it is) sky" and the FINN-R* is top of the FINN-Rs in this figure.

Then the region recognized as sky is interpreted as "wide sky" by FINN-Isky in the following image interpretation process. Finally, the system integrates these results and the rule "If size is very big, color is blue and correlation is high then (it is) wide sky" is output.

Although a rule-based system requires thousands of rules for recognition and interpretation, the proposed system can obtain and modify rules automatically.

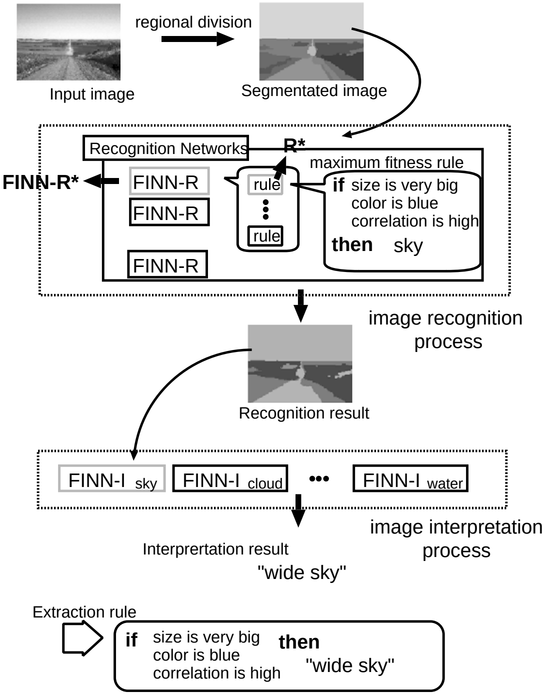


Fig. 10. The flow of rule extraction.

4. Experimental results

4.1. Experimental conditions

Table 4 summarizes the parameters used in the image segmentation process. Structural parameters of FINN-Rs and FINN-Is are shown in Tables 5 and 6, respectively.

We randomly selected 80 images for learning and 8 images are allotted to every FINN-R.

4.2. Regional segmentation result

Figs. 11 and 12 show some results of regional segmentation as a pre-processing. These results are obtained using different parameters for K-means algorithm ($K_p : K_l = 1 : 0.01$) (Fig. 11) and ($K_p : K_l = 1 : 0.3$) (Fig. 12)).

The number under each image indicates the number of regions. It can be observed that the segmentation results after usage of K-means algorithm completely differ by parameters. However, the final segmentation results are similar: the proposed segmentation procedure compensated the difference in parameters.

4.3. Image recognition and interpretation results

Fig. 13 shows some examples of the inference and interpretation results from tested images. In each pair, the left side is an input image and the right side is the recognition result. Numeral under each pair of images indicates the pixel-based recognition rate in which recognition by a human is treated as correct recognition. The sentences are the interpretation results.

Table 4
Parameters of image segmentation

Initial partition of K-means algorithm	3 × 3
Size of Majority filter	5 × 5
$K_l : K_p$	1 : 0.01
S_{th} (%)	0.4
E_{th} (Max : 255)	30
R_{th} (%)	20

Table 5
Parameters of FINN-R

# of FINN-R	10
$N1_{\text{FINN-R}}$	20
$N2_{\text{FINN-R}}$	225 (15 × 15)
$N3_{\text{FINN-R}}$	10
# of Learning images	80 (8 × 10)
$\varepsilon_{self}(t = 0)$	0.5
ε_{LMS}	0.001
$\sigma(t = 0)$	0.5 × 15
# of Learning iterations (self)	2000
# of Learning iterations (LMS)	4000

Table 6
Parameters of FINN-I

# of FINN-I	10
$N1_{\text{FINN}-\text{I}}$	13
$N2_{\text{FINN}-\text{I}}$	400 (20 × 20)
$N3_{\text{FINN}-\text{I}}$	5
# of Learning images	40
$\varepsilon_{self}(t = 0)$	0.5
ε_{LMS}	0.001
$\sigma(t = 0)$	0.5 × 20
# of Learning iterations (self)	2000
# of Learning iterations (LMS)	4000

The average pixel-based recognition rate is 71.9% for 50 test images. It should be noted that there exist many difficult areas to recognize even for humans such as distinction between sky and cloud, that between mountain and wood, etc. In addition, the rate is much higher than the previous study [17] (55.2%).

Each FINN-R (FINN for recognition) learns 10 images and it takes about 60 min on Athlon (1.2 GHz) computers. In the same way, FINN-I (FINN for interpretation) learns the attributes of 40 images and it takes about 30 min on the same computers. Since the learning of FINN-Rs and FINN-Is are based on feature values for each region, the computational time does not depend on the size of images.

4.4. Rule extraction results

The examples of rules obtained from the proposed system are shown in Table 7. These extracted rules are natural and are considered to be correct. The image recognition and interpretation are carried out by these rules.

4.5. Discussion on the usage of multiple FINNs

The proposed system enhances the performance by the usage of multiple FINNs in parallel. Fig. 14 shows the relation between the number of FINN-Rs versus the average rule fitness value $\bar{\rho}$ and the recognition accuracy using 50 test images. Here, $\bar{\rho}$ is the average value of each rule's fitness value ρ (see Eq. (3)) used for each regional recognition. When this value is high, it means that the system has already learnt the similar conditions of the input and that the recognition results are reliable.

According to Fig. 14, it can be confirmed that $\bar{\rho}$ and the recognition accuracy increase as the number of FINN-Rs increases. Since the leaning images for FINN-Rs are randomly selected in the proposed system, if more FINN-Rs are used, recognition accuracy can be increased and that different kinds of images can be treated.

Fig. 15 shows the relation between $\bar{\rho}$ and the recognition accuracy. In this figure, the number of FINN-R

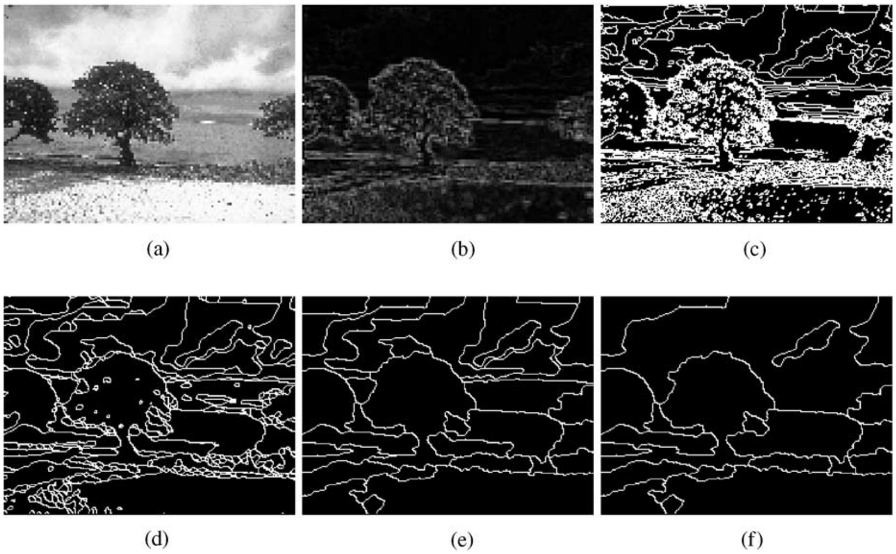


Fig. 11. Examples of segmentation. (a) Original image; (b) Edge image; (c) After K-means algorithm $(K_p : K_l = 1 : 0.01)$: 1785; (d) After majority filter : 273; (e) After small region merging : 16 and (f) After comparison of edges : 13.

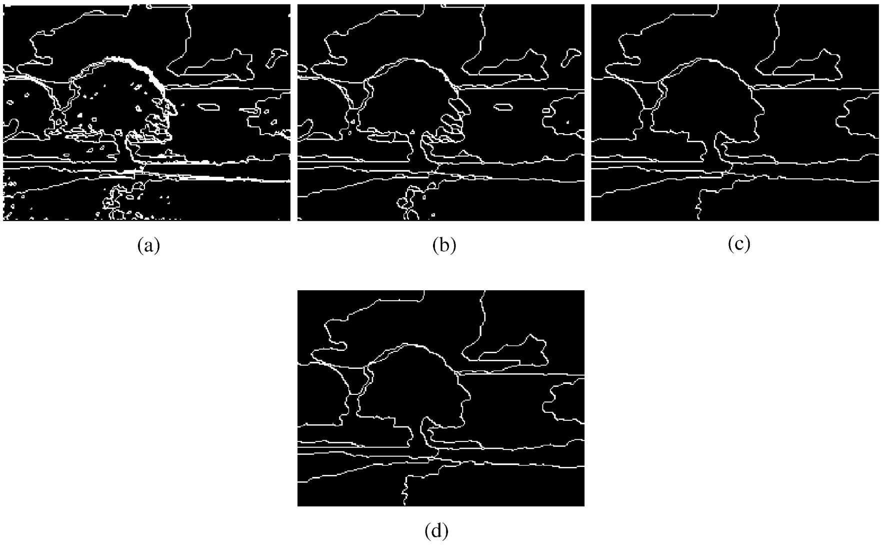


Fig. 12. Examples of segmentation on different parameters. (a) After K-means algorithm $(K_p : K_l = 1 : 0.3)$: 826; (b) After majority filter : 72; (c) After small region merging : 15; (d) After comparison of edges : 15.

was changed from 1 to 10; each FINN-R was tested by 50 images, so the relationship of totally 500 images are plotted in this figure. The "$*$" plot in this figure shows the result by 10 FINN-Rs system ($\bar{\rho} = 0.64$, average recognition accuracy = 71.9%) and the plots located in the larger area of both $\bar{\rho}$ and recognition accuracy. The "$\times$" plot in this figure shows the results by 1-9 FINN-Rs system. From this figure, positive correlation between $\bar{\rho}$ and the recognition accuracy can be observed.

The area where $\bar{\rho}$ exceeds 0.7, average recognition accuracy exceeds 80%. Therefore, it is possible to estimate the reliability of recognition result using the rule fitness value $\bar{\rho}$.

4.6. Analysis of recognition results

Table 8 shows the pixel-based recognition differences between the proposed system and humans. From this

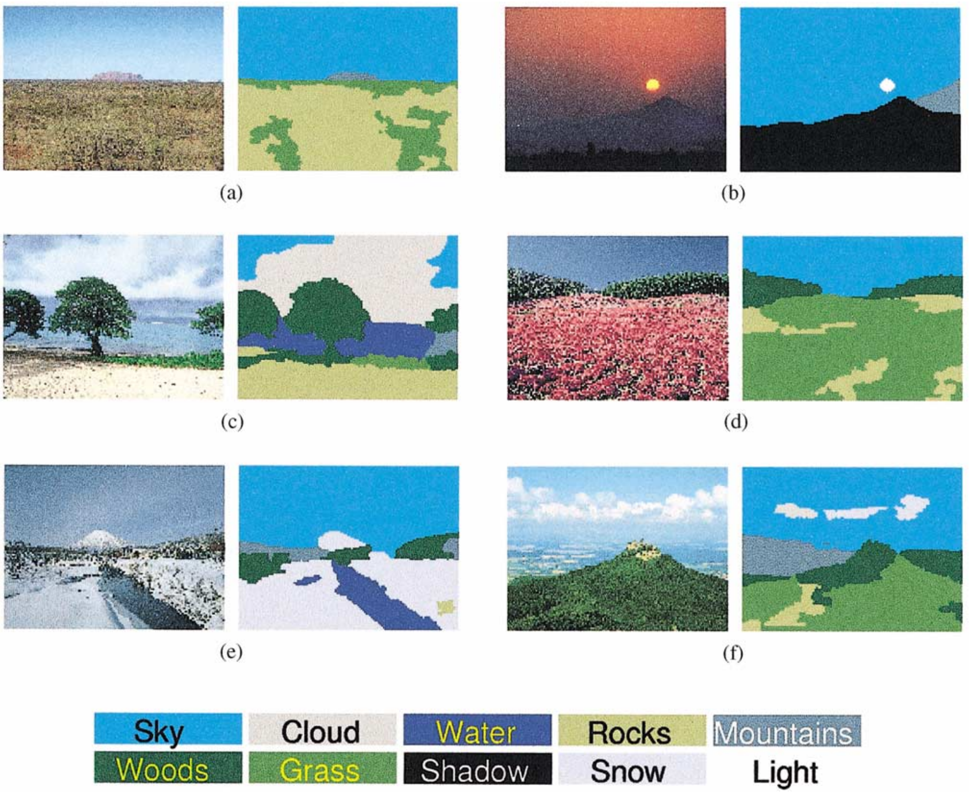


Fig. 13. Examples of image recognition and interpretation. (a) 92.6%: wide sky·grassy plain·far mountain; (b) 97.3%: sunset/night·sun·dark; (c) 90.2%: heavy cloud·ocean·sandy beach·tree; (d) 89.2%: clear sky·many trees·flowers; (e) 75.5%: wide sky·ocean·dead/red tree·covered snow and (f) 49.3%: wide sky·many trees·far mountain.

Table 7
Examples of extracted rules

Ocean	High intensity·light blue· high contrast except in horizontal·high correlation
Tree	Low intensity·green·center area·large entropy·low correlation
Flowers	Rather lower area·pink·low correlation·large entropy
Covered snow	Lower area·high intensity·high correlation in horizontal

table, we can observe that the images used for test include many sky and grass areas. Also, we can see that major errors occur between rock and grass. The errors tend to happen because they have similar characteristics of both the color and the texture. Moreover, extraction of this kind of areas is very difficult in the segmentation process. Then we made a consideration focused on the results based on each label.

Table 9 shows the label-based recognition results. High accuracy of over 90% is obtained on the recognition for sky and shadow areas. In this case, we can think that the system could extract useful characteristics in the segmentation and used them efficiently in the following processes.

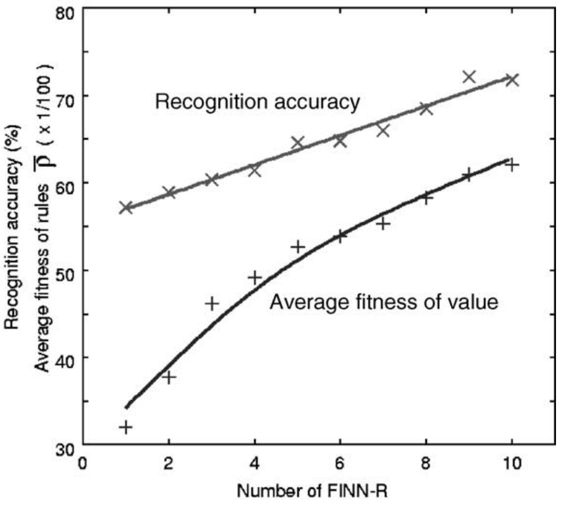


Fig. 14. Number of FINN-R vs. fitness of rules and recognition accuracy.

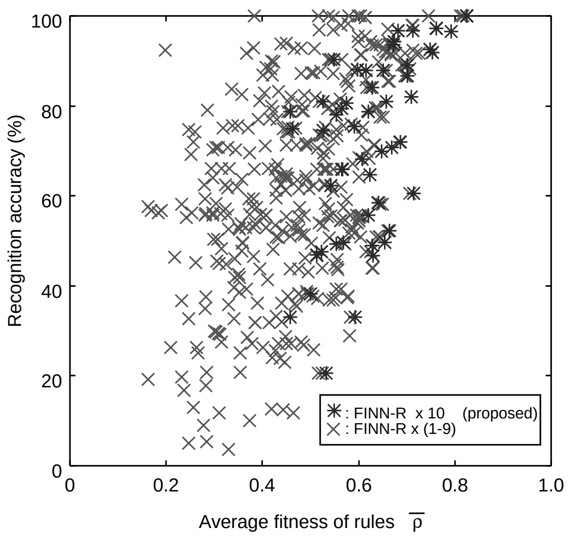


Fig. 15. The relation of fitness of rules and recognition accuracy.

On the other hand, recognition for mountain area is difficult and the accuracy rate is under 40%. Some mountain areas are falsely recognized as woods, water or sky. The exact classification between mountain and woods is sometimes impossible when the mountain is covered with woods. In addition, since mountains in the far distance are often blue in color and some of them have very similar texture tendency with water; false recognition with water happens easily.

As might be mentioned so far, the recognition task itself is much harder compared with a printed character recognition task, for example.

5. Conclusions

In this paper, we have proposed and analyzed a new image recognition and interpretation system using learning mechanism of fuzzy inference neural networks.

In the proposed system, first, segmentation is carried out as a pre-processing. This process that consists of K-means algorithm, majority filtering, small region merging, and comparison with edge image works effectively to segment the image and extract objects.

Table 8
Recognition differences between the proposed system and human (%)

		Recognition by proposed system										
Recognition by human	Sky	28.3	0.7	0.5	0.9	0.1	0.3	0.0	0.0	0.0	0.0	30.8
	Cloud	1.6	5.7	0.0	0.1	0.1	0.2	0.3	0.3	1.5	0.1	9.9
	Water	0.9	0.0	5.5	0.2	0.1	0.2	0.8	0.0	0.5	0.0	8.2
	Mountain	0.7	0.4	0.7	2.1	0.1	1.1	0.6	0.0	0.0	0.0	5.7
	Rock	0.1	0.5	0.4	0.2	6.6	0.7	1.7	0.1	0.8	0.1	11.2
	Wood	0.1	0.1	0.0	0.2	0.8	6.0	2.7	0.0	0.0	0.0	9.9
	Grass	0.0	0.0	0.0	0.0	3.4	1.5	7.9	0.0	0.0	0.2	13.0
	Shadow	0.0	0.0	0.0	0.0	0.1	0.0	0.0	4.9	0.0	0.0	5.0
	Snow	0.0	0.2	0.2	0.0	0.1	0.0	0.0	0.0	3.5	0.0	4.0
	etc.	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.1	1.4	1.7

Table 9
Recognition differences between human and the system (normalized in each label)(%)

		Recognition by proposed system									
		Sky	Cloud	Water	Mountain	Rock	Wood	Grass	Shadow	Snow	etc.
Recognition by human	Sky	90.2	2.3	1.5	2.8	0.2	0.8	0.0	0.0	0.4	1.9
	Cloud	15.8	57.6	0.4	0.8	0.9	1.9	3.0	3.4	15.6	0.5
	Water	10.9	0.6	67.3	2.0	1.1	3.0	9.5	0.0	5.6	0.2
	Mountain	11.9	6.8	12.3	37.0	1.4	19.1	10.6	0.0	0.8	0.0
	Rock	1.2	4.2	3.6	1.4	59.4	6.2	15.7	0.5	7.2	0.5
	Wood	1.5	0.7	0.4	2.2	7.9	60.0	27.4	0.0	0.0	0.0
	Grass	0.0	0.0	0.1	0.2	26.3	11.2	60.6	0.0	0.0	1.5
	Shadow	0.0	0.0	0.0	0.3	1.2	0.0	0.9	97.6	0.0	0.0
	Snow	0.9	4.4	5.5	1.0	1.5	0.0	0.0	0.0	86.3	0.4
	etc.	0.0	0.0	0.0	0.0	14.7	0.0	0.0	0.0	3.8	81.5

As for the image recognition and interpretation processes, multiple modified FINNs are employed to achieve superior results.

The usage of multiple FINNs brings the following merits: the performance is improved; maintenance of small network is easy; additional learning is possible. We have confirmed that the proposed system achieves good results using rules that are obtained automatically in the learning process and that it can extract rules for the image recognition and interpretation.

Acknowledgements

This work is partly supported by the Japan Society for the Promotion of Science ("Research for the Future", Program, JSPS-RFTF 97 I 00103).

References

[1] Y. Ohta, Knowledge-based interpretation of outdoor natural color scenes, Research Notes in Artificial Intelligence, Vol. 4, Pitman, London, 1985.

[2] A.D. Bruce, T.C. Robert, B. John, R.H. Allen, M.R. Edward, The scheme system, Int. J. Comput. Vision 2 (1989) 209–250.

[3] M.S. Thomas, A.F. Martin, Context-based vision: Recognizing objects using information from both 2-D and 3-D imagery, IEEE Trans. Pattern Anal. Syst. 13 (10) (1991) 1050–1065.

[4] J. Yamane, M. Sakauchi, A Construction of a new image database system which realizes fully automated image keyword extraction, IEICE Trans. Inf. Syst. E76-D (10) (1993) 1216–1223.

[5] S. Hirata, Y. Shirai, M. Asada, Scene interpretation using 3-D information extracted from monocular color images, Trans. IEICE, J75-D-II (11) (1992) 1839–1847 (in Japanese).

[6] M. Mirnehdi, Feedback control strategies for object recognition, IEEE Trans. Image Process. 8 (8) (1999) 1084–1101.

[7] J. Peng, B. Bhanu, Closed-loop object recognition using reinforcement learning, IEEE Trans. Pattern Mach. Intell. 20 (2) (1998) 139–154.

[8] M. Mukunoki, M. Minoru, K. Ikeda, A retrieval method of outdoor scenes using object sketch and an automatic index generation method, Trans. IEICE J79-D-II (6) (1996) 1025–1033 (in Japanese).

[9] M. Mukunoki, M. Minoru, K. Ikeda, Retrieval of images using pixel based object models, Proceedings of the Fifth International Conference on Information Processing & Management of Uncertainly, 2 (1994) 1127–1132.

[10] Ashiah Ghosh, Nikhi R. Pal, Sankar K. Pal, Self-organization for object extraction using a multilayer neural network and fuzziness measures, IEEE Trans. Fuzzy Systems 1 (1) (1993) 54–68.

[11] Yoshikazu Nogami, Yoichi Jyo, Michifumi Yoshioka, Sigeru Omatu, Remote data analysis by Kohonen feature map and competitive learning, IEEE Int. Conf. SMC. 1 (1997) 524–529.

[12] Jyh-Shing, Roger Jang, ANFIS: adaptive-network-based fuzzy inference system, IEEE Trans. System, Man, Cybernet. 23 (3) (1993) 665–685.

[13] Li-Xin Wang, Training of fuzzy logic systems using nearest neighborhood clustering, Proceedings of the Second IEEE International Conference on Fuzzy Systems, 1 (1993) 93–100.

[14] Teuvo Kohonen, The self-organizing map, Proc. IEEE 78 (9) (1990) 1464–1480.

[15] Takatoshi Nishina, Masafumi Hagiwara, Fuzzy inference neural network, Neurocomputing 14 (1997) 223–239.

[16] Hiroshi Kitajima, Masafumi Hagiwara, Generalized fuzzy inference neural network using self-organizing feature map, Trans. IEE Japan 117-C (7) (1997) (in Japanese) 971–978.

[17] Hitoshi Iyatomi, Masafumi Hagiwara, Knowledge extraction from scenery image and recognition using fuzzy inference neural network, Trans. IEICE J82-D-II (4) (1999) 685–693 (in Japanese).

[18] S.Z. Selim, M.A. Ismail, *K*-mean-type algorithms, IEEE Trans. Pattern Anal. Mech. Intell. 6 (1) (1984) 81–87.

[19] Shinichi Sakaida, Yoshiaki Shishikui, Yutaka Tanaka, Ichiro Yuyama, An image segmentation method by the region integration using the initial dependence of the *K*-means algorithm, Trans. IEICE J81-D-II (2) (1998) 311–322 (in Japanese).

[20] Tomio Echigo, Shun-ichi Iisaku, Unsupervised segmentation of colored texture images by using multiple GMRF models and hypothesis of merging primitives, Trans. IEICE J81-D-II (4) (1998) 660–670 (in Japanese).

[21] Haruyuki Iwata, Hiroshi Nagahashi, Active region segmentation of color images using neural networks, Trans. IEICE 80-D-II (11) (1997) 2995–3003 (in Japanese).

[22] J.M. Gauch, Image segmentation and analysis via multiscale gradient watershed hierarchies, IEEE Trans. Image Process. 8 (1) (1999) 69–79.

[23] M.L. Comer, E.J. Delp, Segmentation of texture images using a multiresolution Gaussian autoregressive model, IEEE Trans. Image Process. 8 (3) (1999) 408–420.

[24] L.C. Kaplan, Extended fractal analysis for texture classification and segmentation, IEEE Trans. Image Process. 8 (11) (1999) 1572–1585.

[25] Keiko Takahashi, Keiichi Abe, Color Image Segmentation Using ISODATA Clustering Algorithm, Trans. IEICE J82-D-II (4) (1999) 751–762 (in Japanese).

[26] Kazuhiro Kuroda, Masafumi Hagiwara, An image retrieval system by impression words and specific object names—IRIS, Neurocomputing to appear.

About the Author—HITOSHI IYATOMI was born in Tokyo, Japan, on 25 March 1976. He received his B.E. and M.E. degrees in Electrical Engineering from Keio University in 1998 and 2000, respectively. His research interests include image understanding and neural networks. Since 2000 he has been employed by Hewlett Packard Japan.

About the Author—MASAFUMI HAGIWARA is an Associate Professor of Keio University. He was born in Yokohama, Japan, on 29 October 1959. He received the B.E., M.E., and Ph.D. degrees in Electrical Engineering in Keio University, Yokohama, Japan, in 1982, 1984 and 1987, respectively. In 1987, he became a research associate of Keio University. Since 1995, he has been an Associate Professor. From 1991 to 1993 he was a visiting scholar at Stanford University. He received the Niwa Memorial Award, Shinohara Memorial Young Engineer Award, IEEE Consumer Electronics Society Chester Sall Award and Ando Memorial Young Engineer Award in 1986, 1987, 1990 and 1994, respectively. His research interests include soft computing. Dr. Hagiwara is a member of the IEEE, IEICE, IEE of Japan, IPSJ (Information processing society of Japan) and JNNS.