Effect of Grouping in Vector Recognition System Based on SOM

Masayoshi Ohta
Graduate School of
Science and Engineering
Kansai University
Osaka, Japan
Email: k287636@kansai-u.ac.jp

Yuto Kurosaki
Department of
Science and Engineering
Kansai University
Osaka, Japan
Email: k166020@kansai-u.ac.jp

Hidetaka Ito
Faculty of
Engineering Science
Kansai University
Osaka, Japan
Email: h.ito@kansai-u.ac.jp

Hiroomi Hikawa
Faculty of
Engineering Science
Kansai University
Osaka, Japan
Email: hikawa@kansai-u.ac.jp

***Abstract*—This paper discusses effect of grouping on vector classifiers that are based on self-organising map (SOM). The SOM is an unsupervised learning neural network, and is used to form clusters using its topology preserving nature. Thus it is used for various pattern recognition applications. In image recognition, recognition accuracy is degraded under difficult lighting conditions. This paper proposes a new image recognition system that employs a grouping method. The proposed system does the grouping of vectors according to their brightness, and multiple vector classifiers are assigned to every groups. Recognition parameters of each classifier are tuned for the vectors belonging to its group. The proposed method is applied to position identification from images obtained from an on-board camera on a mobile robot. Comparison between the recognition systems with and without the grouping shows that the grouping can improve recognition accuracy.**

I. Introduction

Various pattern recognition systems have been developed that are of practical use, e.g., the personal identification and man-machine interaction. Pattern recognition can be defined as the categorization of input data into identifiable classes, which is a mapping process of the input vectors to a finite set of clusters each of which is associated to a class. The input image is converted to a feature vector and its class is determined by a vector classifier that searches the closest prototype of the cluster.

Many researchers proposed image recognition systems based on a self-organizing map (SOM) [1] that is one of the unsupervised learning neural networks. The SOM is well known for its nonlinear mapping from a given high-dimensional input space to a lower-dimensional map of neurons. One of the important feature of the SOM is vector quantization embedded in the mapping process, with which resembling vectors are mapped to a single neuron. Another interesting nature of the SOM is topological preserving nature, i.e., vectors that are neighbors in the input space will be represented close to each other on the map, too. Due to these features, the SOM has been successfully used for a wide range of applications, such as information visualization and data analysis. Also, the SOM was used for various clustering and pattern recognition applications [2]-[5].

The SOM is used to form clusters, with which the vector quantization is carried out. Then the recognition is carried out using the mapping results. However, the reliability and interpretability of the mapping by the SOM is questionable because the results of the SOM greatly depend on careful selection of parameters like learning rate and neighborhood function that are not intuitive to users [6]. Therefore, it is necessary to interpret the obtained mapping appropriately so that the final classification is made accurately. One of the effective options is to use another SOM. This kind of structure is often referred to as a hierarchical SOM [3]. Another approach is hybrid model with the self-organizing map and supervised neural network [4].

In a hand-sign recognition system [5], a hybrid network called SOM-Hebb classifier consisting of SOM and Hebbian learning network, was employed. The Hebbian-learning network was a single layer feed-forward network, that was used to relate the neurons in the SOM to classes.

In general, the image recognition is highly affected by lighting conditions. For accurate image recognition, the lighting condition during recognition should be the same as that when training images were taken because the prototypes are formed by using the training images. For example, images in the same position taken in the morning and daytime have different brightness. Also the weather condition should be taken into account.

This paper proposes a new type of image recognition system that employs grouping and multiple SOM-Hebb classifiers. Based on their rough brightness pattern, images are allocated to groups, and the images in the same group are classified by the SOM-Hebb classifier that has been tuned for the particular group. The SOM-Hebb classifiers are tuned to different lighting conditions to cope with the variation of the lighting conditions. In order for the system to select the appropriate classifier, another SOM is employed, which performs the grouping of images in terms of their brightness. According to the decision by this SOM, each image is fed to one of the SOM-Hebb classifiers that was tuned to the group to which the image belongs.

In terms of the use of multiple classifiers, a random for-

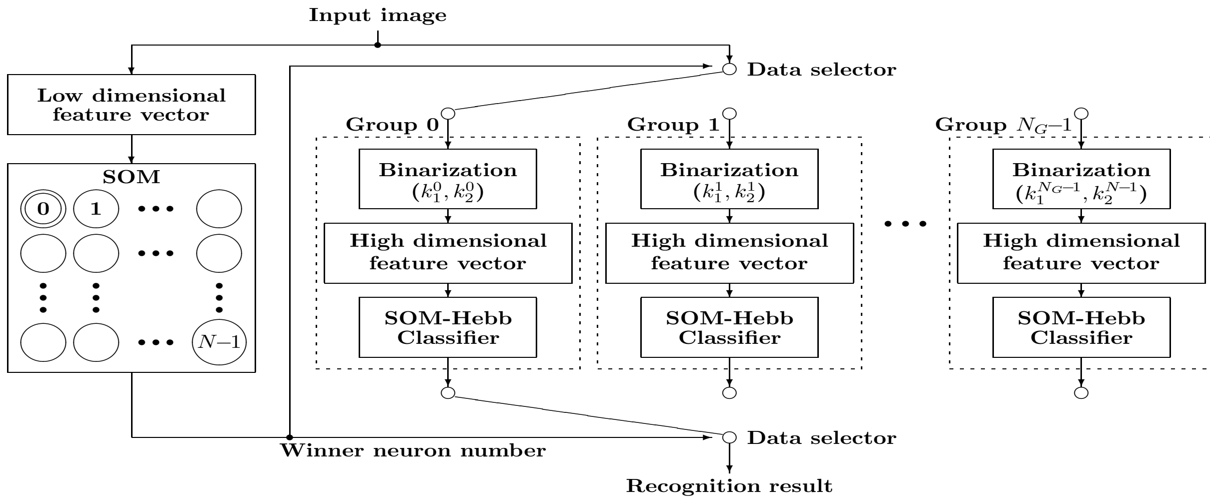

Fig. 1. Group-based recognition.

est [7] employs the similar approach, and is utilized in a wide range of applications, such as image classification and object recognitions. However, the random forest is a collective learning algorithm that develops multiple decision trees, and its result is determined by voting of the trees.

Classification performance of the proposed system is examined by applying it to a position identification, which identifies the position of a camera from its input image. The remainder of this paper is organized as follows: Section II describes the detail of the proposed classifier system by applying it to the position identification application. Then the performance of the system is examined by experiments. The experimental results are given in section III, followed by conclusions in section IV.

II. Recognition using grouping method

A. System summary

Recognition flow of the proposed group-based image recognition algorithm is shown in Fig. 1. The system consists of a SOM and N_G classifiers each of which is tuned to match a specific group of input images. First, a low dimensional feature vector of the input image is extracted and is fed to the SOM. According to the mapping done by the SOM, the input image is forwarded to one of the classifiers, and its recognition result is used as final recognition result. Each classifier is the same one used in hand sign recognition system that was proposed in [5]. The classifier consists of a binarization and a high dimensional feature vector extraction followed by the SOM-Hebb classifier.

The proposed group-based recognition is applied to the position identification system that identifies camera positions by finding a similar landscape with a position data. This similar landscape search is carried out by one of the individual vector classifier. For example, images taken at a certain place in a cloudy day have low brightness, while images taken in a sunny day are very bright. However, the image recognition is highly affected by lighting conditions. The position identification is carried out by comparing input images with pre-memorized images called prototypes. Therefore, it is desirable that the brightness of input image and that of prototype images are close for the accurate recognition.

To solve the problem, in the proposed system, input images are grouped by the SOM in terms of brightness by using the low-dimensional feature vector. Each vector classifier is trained to recognize images in a group that has a narrow range of brightness.

B. Low dimensional feature vector

The input image is given as 24-bit full color, and it is converted to a 256-level gray scale image by using National Television Standards Committee (NTSC) weighted average method, and the formula of the method is given in (1).

$$\begin{aligned} Y(x,y) = &0.298912 \cdot R(x,y) \\ &+ 0.586611 \cdot G(x,y) \\ &+ 0.114478 \cdot B(x,y) \end{aligned} \tag{1}$$

The conversion method forms a weighted average to account for human perception. Since human sensitivity to green is greater than other colors, so green is weighted most heavily. An example of the grayscale conversion is shown in Fig. 2.

After the grayscale transformation, a low-dimensional feature vector is extracted from the grayscale image. Using this low-dimensional vector, the image is grouped. The grayscale image is divided into K sub-images as shown in Fig. 3, where the image is divided into 4 ($K = 4$). Average luminance $A(s)$ of each divided sub-image is computed using the following equation.

$$A(s) = \frac{K \cdot \sum_{y=0}^{Q-1} \cdot \left[\sum_{x=\frac{P}{K}\cdot s}^{\frac{P}{K}(s+1)-1} Y(x,y) \right]}{Q \cdot P}, s = (0, 1, \cdots, k-1) \tag{2}$$

Fig. 2. Grayscale conversion image.

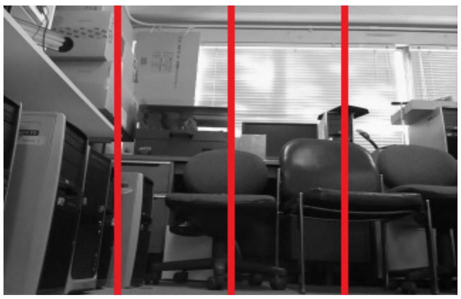

Fig. 3. Example of image division ($K = 4$).

where P is the number of pixels in the horizontal of the image and Q is ones in the vertical. A small value is chosen for the division s so that the $A(s)$ represents rough distribution of brightness in the image. Then the $A(s)$ is converted to frequency domain data by using discrete Fourier transform (DFT). The DFT computation is given in (5).

$$R_S(n) = \sum_{s=0}^{K-1} A(s) \cdot \cos(\frac{2\pi sn}{K}) \tag{3}$$

$$I_S(n) = \sum_{s=0}^{K-1} A(s) \cdot \sin(\frac{2\pi sn}{K}) \tag{4}$$

$$F_S(n) = \frac{\sqrt{{R_S}^2(n) + {I_S}^2(n)}}{K} \tag{5}$$

The resulting frequency domain data $F_S(n)$ is used as the low-dimensional feature vector.

C. Grouping by self-organizing map

The low-dimensional vectors are grouped by using the vector quantization ability of the SOM. Since the low-dimensional vector represents the rough distribution of brightness in the images, the grouping is done on a basis of brightness.

As shown in Fig. 4, the SOM consists of N_G neurons, and each neuron has D-dimensional vector $\vec{m}_i$, called weight vector. Input to the SOM is also a D-dimensional vector, $\vec{x}_i$. Since the SOM is used for the grouping of images using the low-dimensional vectors, $D = K$.

$$\vec{m}_i = (\mu_{i1}, \mu_{i2}, \cdots, \mu_{iK}) \tag{6}$$

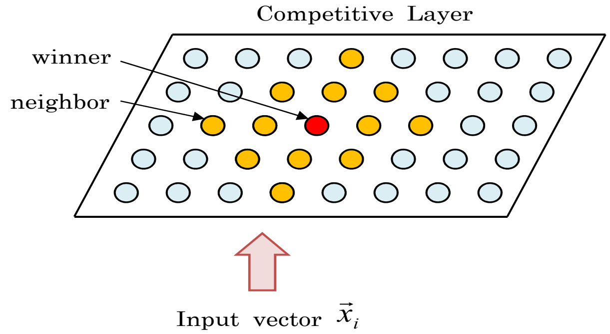

Fig. 4. Self-organizing Map.

$$\vec{x} = (x_1, x_2, \cdots, x_K) \tag{7}$$

Operation of the SOM is divided into two phases, i.e., learning phase and recall phases. In the initial learning phase, the map is trained with a set of input vectors. After the learning phase, the weights of the map are used in the recall phase.

The learning phase starts with an appropriate initialization, in which small random numbers are assigned to the weight vectors. Subsequently, training vectors $\vec{x^t} \in \Re^D$, are fed to the map in multiple iterations. For each training vector, the distances to all weight vectors are calculated, and a *winner neuron* c that has the smallest distance, is determined.

$$c = \arg\min_i \{||\vec{x^t} - \vec{m_i}||\} \tag{8}$$

Euclidean distance is used to measure the distance.

$$||\vec{x} - \vec{m}|| = \sqrt{(\xi_1^t - \mu_1)^2 + (\xi_2^t - \mu_2)^2 + \cdots + (\xi_N^t - \mu_N)^2} \tag{9}$$

After the winner neuron is determined, the weight vectors of the winning neuron and its neighborhood neurons are updated. The vectors are updated by using the following equation, resulting in weight vectors closer to the input vector.

$$\vec{m_i}(k+1) = \vec{m_i}(k) + h_{ci} \cdot (\vec{x^t} - \vec{m_i(k)}), \tag{10}$$

where h_{ci} is a function called neighborhood function. The neighborhood function is defined by

$$h_{ci} = \alpha(k) \exp\left(-\frac{||\vec{p_c} - \vec{p_i}||}{2\sigma^2(k)}\right), \tag{11}$$

where, k is time index. $\vec{p_c} \in \Re^2$ and $\vec{p_i} \in \Re^2$ are the position vectors of the winning neuron c and neuron i respectively. $\alpha(k)$ is a learning coefficient ($0 < \alpha(k) < 1$). $\sigma(k)$ represents the neighborhood radius, and weight vectors within the radius from the winner neuron are updated. Magnitude of the update decreases as the distance to the winner neuron increases.

The learning of the SOM is carried out by repeatedly giving the training vectors, and feature map of the training vector is gradually formed, in which each weight vector of neuron is placed in center of input vector clusters. Therefore the weight vectors can be used as prototype vectors. Using this nature

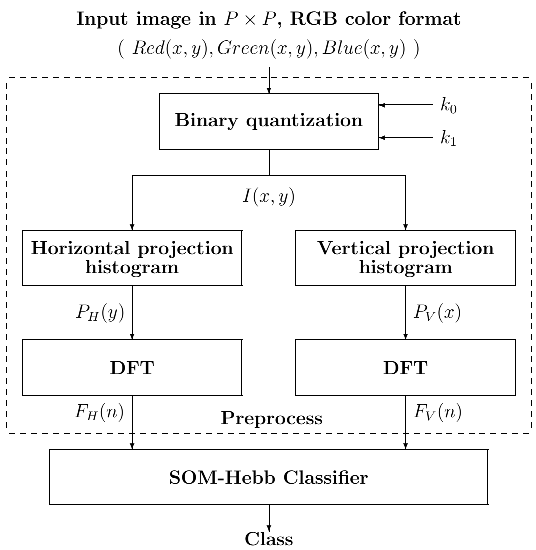

Fig. 5. Recognition flow within each group.

of SOM, the vector quantization is carried out in the recall phase. During the recall phase, only the winner neuron search is conducted. Clusters of the low-dimensional vector space are formed based on the brightness of images. For vectors belonging to one of the clusters, a neuron whose weight vector is placed in the cluster becomes the winner. Therefore, by grouping the images according to the winner neuron, images in the same group are supposed to have similar rough landscape pattern and similar brightness. Consequently, the number of groups is equal to the number of neurons (N_G).

D. Recognition algorithm in groups

This section discusses the recognition algorithm carried out in each group. The input image is assigned to one of the N_G groups, and it is processed in the corresponding classifier. The classifier searches the class of landscape position to which the input image belongs. The process flow for the position recognition used in the proposed system is outlined in Fig. 5, which is basically the same one used in hand-sign recognition system [5].

The input image in RGB color format is preprocessed to generate a high-dimensional feature vector. The preprocessing consists of binary quantization, and horizontal- and vertical-projection histogram calculations that are followed by two DFTs. The DFTs calculate the magnitude spectrum of the histogram data. A D-dimensional feature vector is extracted from the magnitude spectrum, and is fed to the SOM-Hebb classifier that finally identifies the hand posture class.

1) Binarization: The binarization converts an RGB 24-bit color image to a binary image. Equation for the binarization is given in (12).

$$I(x,y) = \begin{cases} 255 & : k_0 - 10 < \frac{G(x,y)}{R(x,y)} \cdot 100 < k_0 + 10, \\ & \text{and} \\ & k_1 - 10 < \frac{B(x,y)}{R(x,y)} \cdot 100 < k_1 + 10 \\ 0 & : \text{otherwise} \end{cases} \tag{12}$$

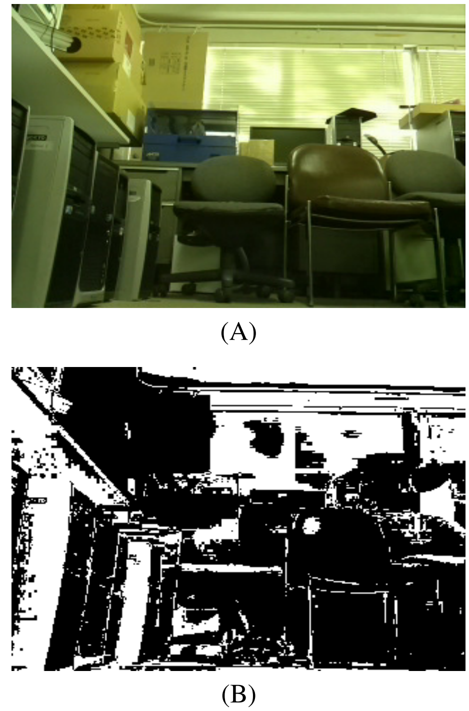

Fig. 6. Binarization example, (A) color image, (B) binarized image.

where, $I(x,y)$ is the binary value at (x,y) coordinates, and $R(x,y)$, $G(x,y)$ and $B(x,y)$ are the color component values of a pixel at the (x,y) coordinates. Using relative values of green and blue against red, the binarized value is selected. Example of the binarization is shown in Fig. 6. Two threshold values k_0 and k_1 must be properly chosen for the system to identify the landscape position of the image. The threshold values are chosen so that a value of 255 is assigned to pixels having reddish color. During the learning, each group searches the best threshold values k_0 , k_1 that maximizes recognition accuracy of the group's classifiers. The appropriate parameters of binarization are used to cope with a change of the lighting conditions.

2) Horizontal and vertical projection histogram : The horizontal and vertical projection histograms of $I(x,y)$, are calculated in the next sub-module. The projection is defined here as an operation that maps a binary image into a one-dimensional array called a histogram. The histogram value is the sum of pixel values along a particular direction. Horizontal projection histogram $P_H(y)$ and vertical projection histogram $P_V(x)$ are defined by

$$P_H(y) = \sum_{x=0}^{P-1} I(x,y), \tag{13}$$

$$P_V(x) = \sum_{y=0}^{Q-1} I(x,y). \tag{14}$$

Fig. 7 shows examples of the horizontal and vertical projection histogram.

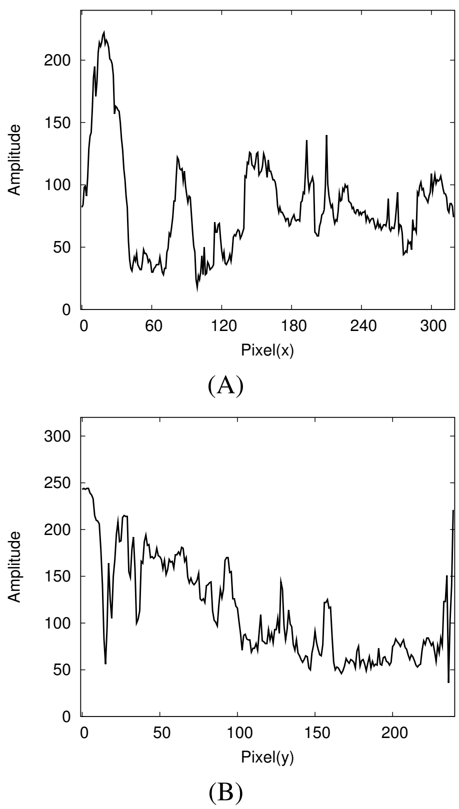

Fig. 7. Histograms of (A) horizontal, (B) vertical.

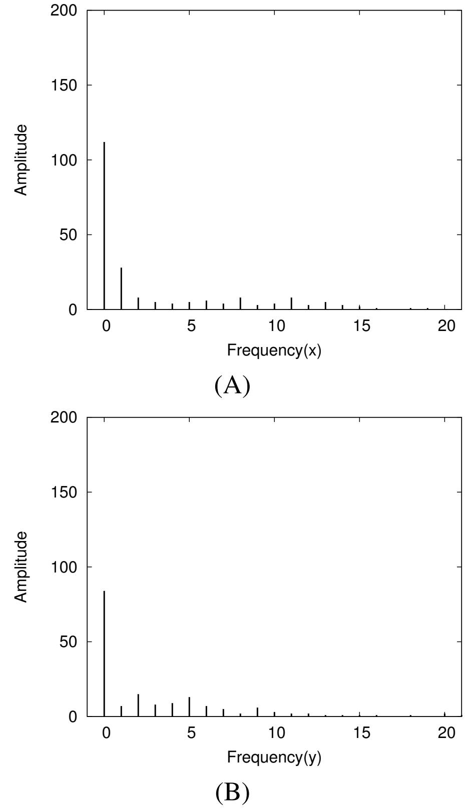

Fig. 8. Frequency spectrum, (A) horizontal, (B) vertical.

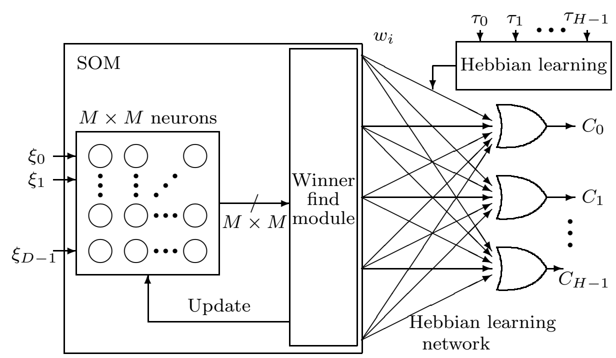

Fig. 9. Hierarchization system using SOM and Hebb learning.

3) Discrete Fourier Transforms : Two DFTs compute the magnitude spectra $F_H(n)$ and $F_V(n)$ of $P_H(y)$ and $P_V(x)$, at the final stage of the preprocessing.

$$A_H(k) = \sum_{n=0}^{Q-1} P_H(n) \cdot \cos(\frac{2\pi nk}{Q}) \tag{15}$$

$$B_H(k) = \sum_{n=0}^{Q-1} P_H(n) \cdot \sin(\frac{2\pi nk}{Q}) \tag{16}$$

$$A_V(k) = \sum_{n=0}^{P-1} P_V(n) \cdot \cos(\frac{2\pi nk}{P}) \tag{17}$$

$$B_V(k) = \sum_{n=0}^{P-1} P_V(n) \cdot \sin(\frac{2\pi nk}{P}) \tag{18}$$

$$F_H(k) = \frac{\sqrt{A_H^2(k) + B_H^2(k)}}{Q} \tag{19}$$

$$F_V(k) = \frac{\sqrt{A_V^2(k) + B_V^2(k)}}{P} \tag{20}$$

Magnitude spectra of $F_H(k)$ and $F_V(k)$ are shown in Fig. 8. As shown in the figure, the lower frequency components have majority of image's feature information. Lower frequency magnitude components obtained by the DFT is used as the feature vector. L-dimensional vector element ξ_i to the classifier network is defined by the following equation.

$$\xi_i = \begin{cases} F_H(i) & 0 \le i < L/2 \\ F_V(i - L/2) & L/2 \le i < L \end{cases} \tag{21}$$

4) SOM-Hebb classifier : The hybrid network consists of SOM, and a single layer feedforward neural network, which is called SOM-Hebb vector classifier. It reads the feature vector and identifies the position where the input image was taken. Fig. 9 shows the SOM-Hebb classifier that identifies H classes. This SOM consists of $N_L = M \times M$ neurons, and dimension of vectors is L. The SOM performs vector classification and the feedforward neural network is employed for class acquisition. Since the feed-forward network is supervised network

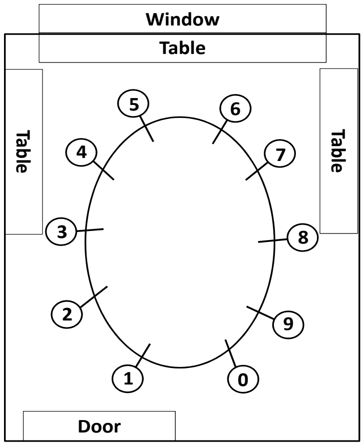

Fig. 10. Recognition points.

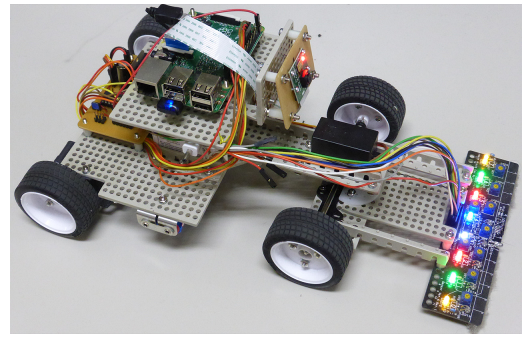

Fig. 11. Mobile robot.

TABLE I
RECOGNITION BY EACH DIVIDED SUB-IMAGE.

Total number of neurons ($N_L \times N_G$)	Number of groups (N_G)	Neurons in group ($N_L = M \times M$)	Number of divided sub-image (K)	Recognition rate [%]
16	1	4×4	2	90.95
			4	92.79
			8	92.17

which is trained by using Hebbian learning algorithm [8], it is called Hebb network in this paper.

As was explained in Sec. II-C, the SOM performs vector quantization. For each input vector, one winning neuron is determined, and the input vector is mapped to the winning neuron that represents a cluster. Therefore, from the winning neuron, the class of the input vector can be identified. Because a single class may consist of multiple clusters, clusters belonging to the same class must be selected so that they are associated with that class. The associations between neurons and classes are done by the Hebb network. During the learning phase, sample vectors and teaching signals $\tau_0, \tau_1, \cdots \tau_{H-1}$ that indicate the class of the given vector, are fed to the network.

A sample vector makes one of the neuron an winner and one of the winner information signal w_k becomes '1'. Then the winner signal activated by the input is connected to the corresponding output node (OR gate) that is indicated by the teaching signal if strong synchronization is found between two signals. The output node may be connected to multiple w_k signals because the class may consist of multiple vector clusters.

If there are too many neurons against the number of classes in the SOM, ineffective neurons that have no connections tend to be formed. Presence of such ineffective neurons with no connection causes failure of classification. If one of the ineffective neurons becomes the winner during classification, no result is given because the winner has no link to any class. To avoid the problem, the ineffective neurons are culled after the training. These ineffective neurons are searched during culling, and their weight vector elements are set to a large value so that they do not become winners.

III. EXPERIMENT

A. Input image

The place where the position identification experiment was performed, is shown in Fig. 10. The robot used to take the image is shown in Fig. 11. The robot runs by using a line trace function on the lane depicted as the oval in Fig. 10. Images were taken in the room at ten points that are numbered $0 \sim 9$, hence the number of classes is $H = 10$. The images varying in both time and weather were taken, and they were transmitted to a computer, on which the proposed recognition was carried out.

B. Setup of experiment

System parameters were set as follows:

- Size of image: $P \times Q = 320 \times 240$
- Number of groups: $N_G = 1, 4, 9$
- Low dimensional vector: $K = 4$
- Total number of neurons in SOM-Hebb: $N_L \times N_G = 16, 36$
- Feature vector dimension: $L = 12$
- Binarization parameters: k_0 and k_1 were searched so that the classification accuracy was maximized.

For the low dimensional vector K was decided by preliminary experiments, results of which is summarized in Table I. 550 images were taken at each position, and 350 of them (3500 in total) were used as the learning data. The remaining 200 images (2000 in total) were used to examine the classification performance of the system. To evaluate effectiveness of the grouping, systems having different number of groups were compared in terms of the classification accuracy. To make the comparison fair, the total numbers of neurons in the systems were all identical.

C. Training

Using the 3500 training images, training of the system was first carried out. The training procedure is given below:

(1) The SOM for the grouping was trained.

TABLE II
k_0, k_1 OPTIMIZATION, WITHOUT GROUPING ($N_L = 16, N_G = 1$).

	k_1										
k0	90	95	100	105	110	**115**	120	125	130	135	140
50	26.67	42.50	41.67	42.94	68.96	64.84	71.04	48.15	58.47	54.37	42.76
55	23.93	43.66	33.59	52.49	85.18	82.45	80.07	44.54	54.09	56.69	48.94
60	18.43	40.20	35.98	49.84	86.85	**90.99**	69.09	37.33	55.03	54.39	45.08
65	10.26	36.30	38.49	47.55	58.68	70.07	44.43	40.17	51.27	46.43	39.97
70	10.00	38.53	41.62	48.77	61.41	61.41	60.13	43.02	51.07	38.81	32.83
75	10.00	42.40	51.87	60.23	69.27	68.85	65.53	40.87	46.33	36.89	31.74
80	10.00	43.95	53.52	64.28	61.73	67.71	50.49	52.29	43.81	33.71	31.63
85	10.00	35.33	54.18	63.43	57.92	55.13	48.01	42.20	35.48	31.62	25.95
90	10.00	47.91	55.54	62.54	59.85	58.92	46.54	36.14	30.37	31.11	22.41
95	10.00	45.37	53.91	47.73	59.09	51.17	42.98	34.33	28.15	23.60	7.98
100	9.18	45.21	42.03	45.07	48.24	30.12	20.83	16.19	15.94	7.41	3.47

(2) Using the above SOM, training images were grouped into N_G.

(3) Each SOM-Hebb classifier was trained using the above grouped training images.

(4) For all individual classifiers, their recognition rates were obtained by using their training images that had been assigned to them.

(5) With different k_0 and k_1, (2) $\sim$ (4) were repeated to find the best pair of k_0 and k_1.

Tab. II shows recognition rates of the system with $N_G = 1$ and $N_L = 16$, i.e., recognition system without the grouping. Therefore, this system used a single SOM-Hebb classifier with 16 neurons ($N_L = 16$). This result shows that $k_0 = 60$ and $k_1 = 115$ provided the best performance.

Tab. III shows training results of the system that consisted of four groups, and each group used 4 neurons in their SOM-Hebb classifiers ($N_L = 2 \times 2 = 4$). Note that total number of neurons in this system ($N_L \times N_G$) was 16, which is identical to that in the system discussed above.

Tab. III shows relations between the pairs of k_0, k_1 and the recognition accuracies of the SOM-Hebb classifiers. Numbers in bold indicate the best pair of k_0 and k_1. Using the same procedure, the best k_0, k_1 for the system with 36 neurons ($N_L \times N_G = 36$) were obtained. Using the best parameters all classifiers were configured.

D. Recognition experiment

Experimental results of the systems with different number of groups are summarized in Tab. IV that shows the recognition rates systems with N_G=1 (no grouping) and N_G=4 (with grouping). The table summarizes individual groups and the overall recognition rate. M_G is the number of test data that were assigned to each group, and M_C is the number of test data that were correctly identified.

Results in the table show that the system with grouping provides better recognition accuracy compared to that of the system without the grouping. Comparison between two systems with the same number of neurons show that the grouping method provided the better performance. In case of 16 neurons, by employing the grouping recognition accuracy was improved by 9.49%. Recognition rate of the system with 64 neurons were improved from 95.92% to 99.38%. Also note that computing time to complete all test was greatly reduced. Recognition rate has been shown to improve the 9.49% in the case of the total number of neurons 16 in Tab. IV. And the grouping method achieves recognition rates of up to 99.38%.

TABLE III
k_0, k_1 OPTIMIZATION, WITH GROUPING ($N_L = 4, N_G = 4$),
(A) GROUP 0, (B) GROUP 1, (C) GROUP 2, (D) GROUP 3.

(A)

	k_1										
k0	90	95	100	105	110	**115**	120	125	130	135	140
50	41.30	63.35	65.71	73.98	81.87	81.71	81.56	71.14	64.46	56.59	51.16
55	37.05	53.12	65.30	67.89	81.90	81.90	81.90	68.97	74.48	78.86	73.12
60	36.67	39.39	63.71	76.84	88.44	**88.57**	81.31	45.60	58.88	78.15	66.02
65	39.10	35.54	61.98	65.03	86.67	87.85	80.86	64.97	74.25	70.48	54.38
70	39.43	38.63	63.92	66.34	66.36	68.38	81.49	65.71	65.73	65.66	61.12
75	6.67	41.07	51.87	75.49	71.14	74.04	81.79	68.42	66.72	65.70	62.76
80	13.16	54.27	51.98	68.80	66.10	82.06	85.70	77.83	68.70	65.62	63.14
85	6.67	59.03	59.05	71.98	47.41	74.36	76.08	65.83	65.70	64.13	57.62
90	6.67	74.36	72.19	71.90	64.17	72.29	69.94	65.41	65.31	58.90	47.70
95	6.67	76.19	75.81	75.47	68.57	62.53	63.09	61.73	59.24	45.54	34.99
100	6.67	72.23	71.70	68.23	69.10	60.82	52.36	49.03	39.98	35.20	13.22

(B)

	k_1										
k0	90	95	100	105	**110**	115	120	125	130	135	140
50	39.86	59.17	46.25	62.51	67.43	58.25	62.23	63.89	73.89	59.15	46.00
55	21.70	65.66	46.25	70.21	89.43	72.23	72.41	63.08	75.24	64.53	48.51
60	34.80	64.69	52.87	67.52	**93.59**	89.24	71.93	62.76	64.94	65.17	49.06
65	39.06	61.79	36.11	70.00	91.49	85.17	90.11	58.25	62.71	58.71	42.60
70	28.21	64.37	48.32	61.75	79.91	84.07	65.89	62.21	50.09	39.08	40.16
75	11.79	69.24	56.39	81.56	76.34	76.87	64.16	57.70	48.60	44.32	36.44
80	11.26	68.11	66.11	80.55	83.36	71.93	54.62	54.30	49.77	44.14	6.41
85	20.16	63.61	75.52	67.52	77.54	63.22	55.43	41.59	42.87	43.43	1.77
90	13.86	82.41	76.60	90.60	81.49	72.00	57.61	49.10	43.43	21.86	0.76
95	4.90	83.98	84.32	84.14	84.92	75.59	66.23	46.90	41.56	6.97	1.36
100	0.64	67.49	73.93	74.83	76.32	53.82	43.24	40.23	1.59	0.41	1.77

(C)

	k_1										
k0	90	95	100	105	**110**	115	120	125	130	135	140
50	48.27	66.22	67.05	52.24	59.91	79.31	72.61	50.97	47.17	53.11	58.18
55	49.40	52.68	61.00	65.48	72.67	72.68	52.53	46.23	57.06	62.01	58.98
60	36.44	56.19	53.41	79.01	**94.84**	89.72	73.94	50.53	58.71	52.35	59.42
65	49.11	53.39	56.11	64.93	65.62	65.51	71.22	48.65	54.54	60.09	53.42
70	49.31	58.89	58.02	51.82	65.78	64.03	68.30	55.26	44.44	52.64	45.54
75	49.45	56.64	60.55	64.57	72.67	62.10	64.05	63.41	40.46	47.90	43.20
80	49.45	74.19	75.35	68.79	77.43	65.59	65.31	63.09	40.39	45.10	41.79
85	21.98	72.95	75.35	68.19	61.26	52.12	51.62	41.11	32.78	37.76	37.57
90	49.45	53.34	54.54	55.59	50.27	45.62	36.67	37.25	36.86	38.76	39.56
95	49.45	51.02	51.30	51.41	52.94	47.52	45.87	43.34	40.02	38.59	33.49
100	17.83	49.14	49.45	49.45	49.45	46.41	38.54	10.06	31.19	3.78	7.83

(D)

	k_1										
k0	90	95	100	105	110	**115**	120	125	130	135	140
50	6.54	8.50	89.54	86.21	98.04	97.14	96.04	91.50	88.76	90.13	91.83
55	6.54	6.54	95.10	90.33	96.73	**98.04**	98.04	91.44	91.50	91.96	93.66
60	6.54	6.86	85.49	90.00	93.46	91.50	98.04	82.68	81.37	89.80	94.25
65	0.00	8.17	93.53	80.00	82.81	87.78	90.20	88.43	88.63	78.43	84.90
70	0.00	8.82	88.89	97.97	91.50	94.77	93.46	91.63	89.15	78.43	73.53
75	0.00	8.95	92.09	98.04	83.20	91.50	94.77	93.46	89.02	78.43	36.80
80	0.00	6.73	91.31	81.76	84.97	91.50	91.50	90.85	86.80	78.43	69.41
85	0.00	98.04	97.84	87.65	80.20	78.43	78.43	78.43	78.43	78.43	80.65
90	0.00	13.07	19.61	98.04	98.04	98.04	91.37	84.38	81.37	78.43	78.43
95	0.00	13.07	13.07	14.38	19.61	51.96	39.80	77.84	78.43	78.43	78.43
100	0.00	13.07	13.07	13.07	13.07	13.07	8.76	15.62	78.43	78.43	78.43

TABLE IV
RECOGNITION RESULT.

Total number of neurons ($N_L \times N_G$)	Number of groups (N_G)	Neurons in group ($N_L = M \times M$)	K_0	K_1	M_G	M_C	Individual group recognition rate [%] ($100 \times M_C/M_G$)	Recognition rate [%] ($100 \times \sum M_C / \sum M_G$)	Recognition processing time [s]
16	1	4×4	60	110	2000	1709	85.46	85.46	9.485
	4	2×2	60	115	613	552	90.01	94.95	2.464
		2×2	60	110	447	435	97.40		
		2×2	60	110	700	672	95.97		
		2×2	55	115	240	240	100.0		
36	1	6×6	60	110	2000	1938	95.92	95.92	17.956
	9	2×2	80	105	20	20	100.00	99.38	2.512
		2×2	65	115	324	324	100.00		
		2×2	60	110	266	264	99.25		
		2×2	65	110	358	357	99.72		
		2×2	65	115	70	70	100.00		
		2×2	75	110	290	280	96.55		
		2×2	70	115	95	95	100.00		
		2×2	55	120	167	167	100.00		
		2×2	60	115	410	410	100.00		

IV. CONCLUSION

This paper proposed a new image classification method, which performs grouping the input vectors based on their rough feature. The method employed multiple classifiers, each of which takes care of the recognition within an assigned group. Using the images allocated to a group, one of the classifier was trained so that its classification performance for the images in that group was maximized. After the training the proposed system had vector classifiers, each of which had been customized for the assigned group. Individual vector classifiers employed the SOM-Hebb classifier.

The proposed method was applied to the position identification. The position identification system identifies the position where its input image was taken, by comparing the image with pre-trained prototype images.

To examine the effect of the grouping, comparison was made between systems with different group configurations and the same number of neurons in the SOM-Hebb classifiers. The experiment showed that the recognition accuracy was improved. This paper revealed that the proposed classifier with the grouping and multiple classifiers has better classification accuracy than that of system without the grouping.

REFERENCES

[1] T. Kohonen, *Self-Organizing Maps*, 3rd, ser. Springer Series in Information Sciences 30. New York: Springer-Verlag, 2001.

[2] T. Kurita, H. Shimai, T. Mishima, and T. Takahashi, “Self-Organization of viewpoint dependent face representation by the self - supervised learning and viewpoint independent face recognition by the mixture of classifiers,” *Proc. IAPR Workshop on Machine Vision Applications*, pp.319-322, 2000.

[3] J. Lampinen, E. Oja. “Clustering properties of hierarchical self-organizing maps,” *Journal of Mathematical Imaging and Vision*, pp. 261-272, 1992.

[4] T. Riga, A. Cangelosi, A Greco, “Symbol Grounding Transfer with Hybrid Self-Organizing/Supervised Neural Networks,” *Proc. IJCNN’04*, pp.2865-2869, 2004.

[5] H. Hikawa, K. Kaida, “Novel FPGA Implementation of Hand Sign Recognition System with SOM-Hebb Classifier” *IEEE Trans. on Circuits and Systems for Video Technology*, vol.25, no.1, 2015.1.

[6] S. S. Shahapurkar, M. K. Sundareshan, “Comparison of Self-Organizing Map with K-Means Hierarchical Clustering for Bioinformatics Applications,” *Proc. IJCNN’04*, pp.1221-1226, 2004.

[7] L. Breiman, “Random Forests” *Machine Learning, Vol.45*, pp.5-32, 2001.

[8] D. O. Hebb, *The Organization of Behavior: A Neuro-physiological Theory*, Wiley, New York, 1949.