Regular Paper

Euro Banknote Recognition System
Using a Three-layered Perceptron and RBF Networks

Masato Aoba,[†] Tetsuo Kikuchi[††] and Yoshiyasu Takefuji[†††]

We propose an Euro banknote recognition system using two types of neural networks; a three-layered perceptron and a Radial Basis Function (RBF) network. A three-layered perceptron is well known method for pattern recognition and is also a very effective tool for classifing banknotes. An RBF network has a potential to reject invalid data because it estimates the probability distribution of the sample data. We use a three-layered perceptron for classification and several RBF networks for validation. The proposed system has two advantages over the system using only one RBF network. The feature extraction area can be simply defined, and the calculation cost does not increase when the number of classes increases. We also propose to use infra-red (IR) and visible images as input data to the system since Euro banknotes have quite significant features in IR images. We have tested our system in terms of acceptance rates for valid banknotes and rejection rates for invalid data.

1. Introduction

Many kinds of banknote recognition machines are available in our society. They are very useful devices to free people from bothering jobs including counting banknotes, changing money or vending tickets. On the other hand, it is a fact that the recent advancement of copy machines or scanners enables us to duplicate counterfeit banknotes. While some recent copy machines and scanners recognize banknotes for rejecting them [1,2], while old ones do not have such function. Therefore, a demand for banknote recognition machines for rejecting counterfeit banknotes has been growing.

Banknote recognition systems using neural networks to classify known banknotes have been reported in some papers [3]~[5]. Takeda et al. proposed the use of multi-dimensional Gaussian probability function for rejecting unknown banknotes [6]. In most cases, it is difficult to estimate the data distribution by Gaussian function. In some patents, validation methods using neural networks are introduced. Eccles reported that a Probabilistic Neural Network (PNN) can reject some unknown data [7]. The PNN has a problem that the larger network size, the larger the number of given classes increases. Baudat proposed a system using a Learning Vector Quantization (LVQ) network [8]. While it solves the network size problem in a PNN, setting thresholds to reject unknown data or outliers is very difficult since an LVQ network originally makes Voronoi diagram for classification.

Broomhead and Lowe introduced Radial Basis Function (RBF) network in 1988 [9]. The RBF network model approximates data distribution by probability distribution and it can reject invalid data. Some papers reported authorization system by using RBF network [10,11], but any banknote recognition system using this neural network model has never been reported yet.

In this paper, we propose a banknote recognition system composed of two parts; a classification part and a validation part. The classification part uses a three-layered perceptron and the validation part uses several RBF networks. While the three-layered perceptron is a well known method for pattern recognition [12]~[14] and is also very effective for classifing banknotes, it makes boundaries only for classifing given training data and is unassured of rejecting unknown data. As mentioned above, the RBF network has a data approximation property, which seems a proper tool for rejecting unknown data.

It is able to configurate the system employing only one RBF network. If we consider an one-phased RBF network method, it has two problems. First, defining feature extraction areas in image data is quite complicated. Second, the caclulation cost is in proportion to $O(mn^2)$,

  † Graduate School of Media and Governance, Keio University
 †† Software Service Department, TOYO Communication Equipment Co., Ltd.
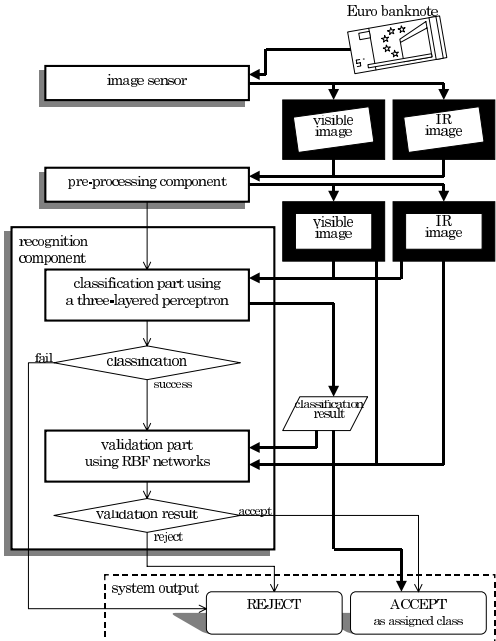††† Faculty of Environmental Information, Keio University

Fig. 1 Overview of the system.

Fig. 2 Overview of the classification part.

where m is the number of kernels for each class and n is the number of given classes. We have designed two-phased method for solving these problems. The feature extraction area can be simply defined for each class. In addition, the calculation cost of the validation part is independent of the number of given classes. More details of these topics are discussed in Section 7.

We apply our idea to Euro banknote recognition system and also propose using infra-red (IR) and visible images as input data to the system since Euro banknotes have quite significant features in IR images.

2. Overview of the System

The overview of the proposed system is shown in **Fig. 1**.

The input data to the system are obtained by an image sensor that has a green LED and an IR LED. The sensor alternately lights a green LED and an IR LED to get a visible image and an IR image. Once the image sensor obtains image data, the pre-processing component detects edges of the banknote to transform images into the right position and angle.

The recognition component obtains a pre-processed image as an input and outputs a recognition result. The recognition component is divided into two parts; a classification part
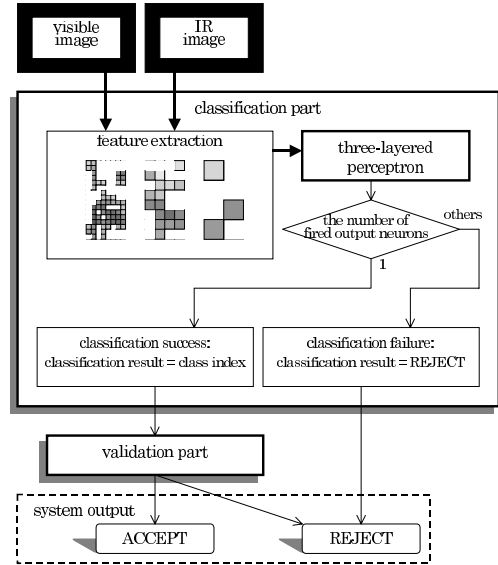
and a validation part.

In the classification part, a previously trained three-layered perceptron is implemented and it requires a visible image and an IR image as input data. If an inserted banknote is classified into a certain class, the result is inputted to the validation part. Otherwise (in case the classification is failed), the validation part is skipped and the system outputs "REJECT". More details of the classification part are described in Section 3.

In the validation part, several trained RBF networks are implemented and they require a visible image and an IR image as input data. The validation part outputs "ACCEPT" or "REJECT". If the validation result is "ACCEPT", the system outputs the classification result. Otherwise (in case the validation result is "REJECT"), the system outputs "REJECT". More details of the validation part are described in Section 4.

3. Classification Part

The overview of the classification part is shown in **Fig. 2**. In the classification part, a three-layered perceptron is employed. The structure of a three-layered perceptron is described in Section 3.1. The visible image and the IR image are transformed into multiresolutional images, and significant input values are selected from the multiresolutional images as input vector to the three-layered perceptron. The method selecting significant input values
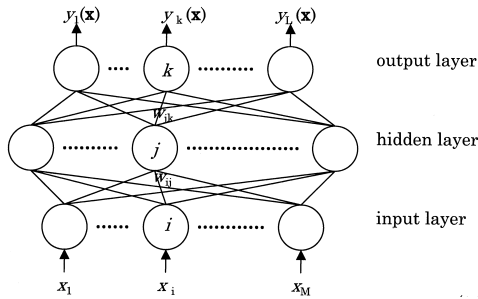
Fig. 3   Structure of a three-layered perceptron.

Fig. 4   Concept of the reduction of input neurons.

is described in Section 3.2. The three-layered perceptron outputs the result of the classification. In case the output of the three-layered perceptron indicates that the inserted banknote is one of the given classes, the banknote is labelled as the class index and the index is inputted to the validation part. Otherwise (in case the output of the three-layered perceptron shows that the inserted banknote cannot be assigned to any given classes or can be assigned to more than two classes), the classification result is "REJECT", then the validation part is skipped and the final output of the system is "REJECT".

3.1   Three-layered Perceptron

A multi-layered perceptron is a kind of feedforward neural networks which is well known tool for pattern recognition. In our system, a three-layered perceptron is employed in the classification part.

The structure of a three-layered perceptron is shown in **Fig. 3**. A three-layered perceptron is composed of an input layer, a hidden layer and an output layer. Each input neuron is fully connected to the hidden neurons and each hidden neuron is fully connected to the output neurons. The strength of the connection between neuron i and j is represented by weight value w_{ij}. The output of each neuron is calculated by the sigmoidal function and the input of the sigmoidal function is the sum of products of the output values and the weight values from the previous layer. The output of the neuron j and a sigmoidal function are represented by

$$y_j = f\left(\sum_i w_{ij}y_i\right) \qquad (1)$$

$$f(x) = \frac{1}{1 + \exp(-x)} \qquad (2)$$

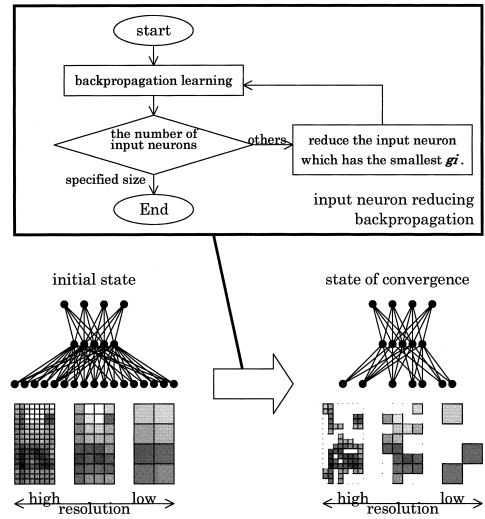where $f(x)$ is the sigmoidal function and y_j is the output of the neuron j.

The input value of each input neuron corresponds to a component of an input vector $\mathbf{x}$ and each output neuron corresponds to each class index.

3.2   Multiresolutional Input Values and Reduction of Input Neurons

The input data to the classification part are extracted from rectangular central area of the inserted banknote. The size of the available area depends on the size of the smallest banknote. Multiresolutional input values extracted from the visible image and the IR image are used as the input vector $\mathbf{x}$ to the three-layered perceptron since image data has significant features in various resolutions.

Note that if all of the multiresolutional input values are used as input data, the dimension of an input vector becomes a huge number. The dimension of the input vector should be reduced because it causes a difficulty of parameter estimation and the problem is well known as "curse of dimensionality"[15]. Actually, some of the input values are not significant features for classification and they are able to be ignored. Matsunaga et al. proposed a learning method to remove redundant hidden neurons[16] and we improve the method in order to remove redundant input neurons.

The concept of reduction of redundant input neurons is shown in **Fig. 4** and its procedure is explained as follows.

At first, the three-layered perceptron is trained by backpropagation method[21]. Consider i and j are indices of the neurons, where the neuron i is the previous neuron of the neu-

ron j. The generalized delta rule is defined as following equation,

$$\Delta w_{ij} = \eta \delta_j y_i \qquad (3)$$

where δ_j is the error value at neuron j, y_i is the output of the neuron i, and η is the learning rate.

If the neuron j is an output neuron, δ_j in Eq. (3) is written as

$$\delta_j = (t_j - y_j)f'(net_j) \qquad (4)$$

where $f'(x) = df(x)/dx$ and t_j is the target signal for the neuron j.

If the neuron j is a hidden neuron, δ_j is given by

$$\delta_j = f'(net_j) \sum_k \delta_k w_{kj} \qquad (5)$$

where net_j is the sum of products described by following equation.

$$net_j = \sum_k w_{jk} y_k \qquad (6)$$

Once the network converges to a stable state by the backpropagation method, the cost g_i is calculated for each input neuron i,

$$g_i = \sum_p \sum_j z_{ip} w_{ij} \qquad (7)$$

where j is the index of the hidden neuron, p is the index of the input patten and z_{ip} is the output value of the input neuron i for the patten p. The value of g_i indicates the significance of the input neuron i for classification in the network. Therefore, if the input neuron i has a great effect on classification, g_i has a large value. Conversely, the input neuron i whose g_i has the smallest value among all input neurons is the most redundant neuron, and that neuron can be removed.

Until the number of the input neurons becomes a certain number, the backpropagation learning and the reducing redundant input neurons is repeated.

4. Validation Part

The overview of the validation part is shown in **Fig. 5**. In the validation part, the visible image and the IR image are divided into some small areas. The validation part is composed of some validation blocks and each validation block corresponds to each given class. A validation block has several RBF networks and each RBF network corresponds to each small area in a image datum. For example, in case the number of classes is L and the image data are divided into K small areas, the validation
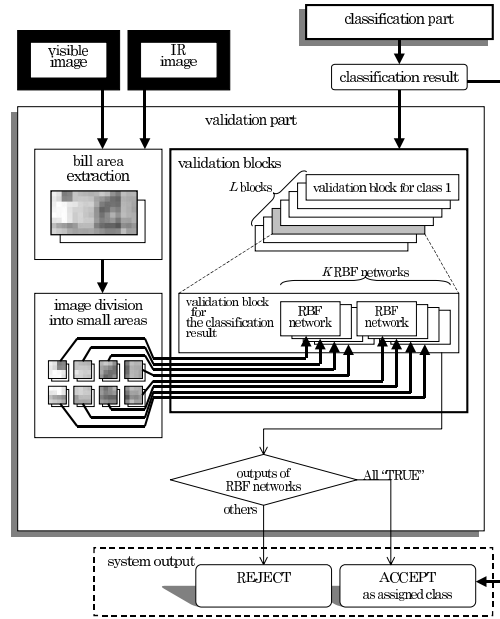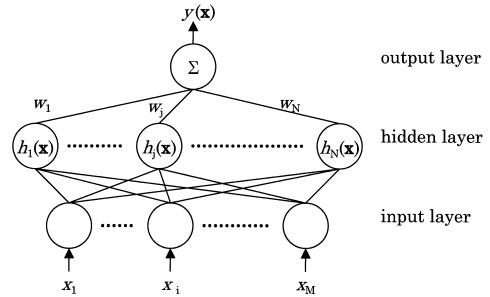

Fig. 5 Overview of the validation part.

Fig. 6 Structure of an RBF network.

part is composed of L blocks, each of which has K RBF networks for each small area. The validation part selects the proper block corresponds to the classification result. Each small area of the image data is transformed into the input vector to assigned RBF network, and the network outputs the validation result for that small area. Only if the all outputs of the RBF networks in the selected block are "TRUE", the output of the validation part is "ACCEPT" and the system outputs the classification result as the final output. Otherwise, the output of the validation part is "REJECT" and the final output of the system is "REJECT".

4.1 RBF Network

A Radial Basis Function (RBF) network [9] is a three-layered network that consists of an input layer, a hidden layer and an output layer. The structure of an RBF network is shown in **Fig. 6**. Each input neuron corresponds to a

component of an input vector $\mathbf{x}$. Each hidden neuron calculates a kernel function which is usually defined by Gaussian function. Here we define the kernel function $h_j(\mathbf{x})$ for the hidden neuron j by the following equation,

$$h_j(\mathbf{x}) = \exp(-(\mathbf{x} - \mathbf{c}_j)^T \mathbf{R}_j(\mathbf{x} - \mathbf{c}_j)) \quad (8)$$

where $\mathbf{c}_j$ is the center of the hidden neuron j. $\mathbf{R}_j$ is a $M \times M$ diagonal matrix defined by kernel width vector σ_j as

$$\mathbf{R}_j = \text{diag}\left(\frac{1}{2\sigma_{j1}^2}, \ldots, \frac{1}{2\sigma_{jM}^2}\right) \quad (9)$$

where M is the dimension of the input vector $\mathbf{x}$.

In a general RBF network, the output layer has more than two neurons for classification. In our system, only one output neuron is necessary for each RBF network in the validation part because the RBF networks in this part do not perform classification process. Only each RBF network has to do is to validate each small area. Here we consider the special case that an RBF network has only one neuron in the output layer.

An output neuron is fully connected to the hidden layer and the output value is calculated by

$$y(\mathbf{x}) = \sum_{j}^{N} h_j(\mathbf{x})w_j \quad (10)$$

where N is the number of the hidden neurons in the hidden layer and w_j is the weight between the hidden neuron j and the output neuron.

Some learning methods for an RBF network were also reported [17),18)]. Schwenker, et al. reported that the three-phase learning with Learning Vector Quantization (LVQ) is the best learning method for an RBF network [19)]. Note that LVQ is equivalent to Self-Organizing Map (SOM) [20)] in our system since the RBF networks in the validation part have only one output neuron. We use the three-phase learning using SOM.

In the first phase, the centers of the hidden neurons are adjusted by SOM. The learning rule of SOM at the time t is described by

$$\Delta \mathbf{c}_{j^*} = \alpha(t)(\mathbf{x} - \mathbf{c}_{j^*}) \quad (11)$$

where $\mathbf{c}_{j^*}$ is the nearest hidden neuron vector to the input $\mathbf{x}$ and $\alpha(t)$ is the learning rate. For convergence, $\alpha(t)$ should have a small positive value and decrease by degree to zero as the time t grows up.

In the second phase, the kernel widths and output weights are temporarily determined. The components of kernel width vector σ_j are set as the average of the distance to the p nearest hidden neuron vectors of $\mathbf{c}_j$. After that, the output weights are adjusted by delta learning rule

$$\Delta w_j = \eta h_j(\mathbf{x})(y - F) \quad (12)$$

where η is a learning rate and F is the target signal.

In the third phase, backpropagation learning is applied to improve the performance of the network. Parameters w_j, $\mathbf{c}_j$ and σ_j are readjusted by the following rules:

$$\Delta w_j = \eta h_j(\mathbf{x})(y - F) \quad (13)$$

$$\Delta c_{jk} = \eta h_j(\mathbf{x})\frac{x_k - c_{jk}}{\sigma_{jk}^2} w_j(y - F) \quad (14)$$

$$\Delta \sigma_{jk} = \eta h_j(\mathbf{x})\frac{(x_k - c_{jk})^2}{\sigma_{jk}^3} w_j(y - F) \quad (15)$$

where η is a learning rate and k is the index of the component of the input vector $\mathbf{x}$.

4.2 Feature Extraction and Dividing into Small Areas

The visible image and the IR image are used as input data to the validation part. The available area is extracted from the image data. It is defined as a rectangular area that does not contain the vicinity of the banknote edges since the features around the edges are affected by wrinkles, creases, cuts, and so on. The size of the available area depends on the banknote class which the classification result indicates. Moreover, the available area is divided into the same sized small areas and each small area is transformed into the input vector $\mathbf{x}$ to each RBF network. Each input vector are normalized in each area. Dividing an available area into some small areas aims at the effectiveness of using local features and at avoiding "curse of dimensionality". That is, if the available area is not divided, the input dimension of an RBF network in the validation block becomes large and the network becomes irresponsive to local features.

5. Training Conditions

The system is designed for recognizing all kinds of Euro banknotes (EUR 5, 10, 20, 50, 100, 200 and 500). We use 200 pieces of new banknotes for each kind of Euro banknotes for training. The size of the available area in image data for the classification part is defined as 58 [mm]×110 [mm] and it depends
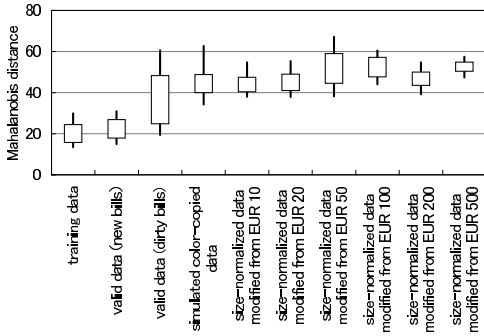
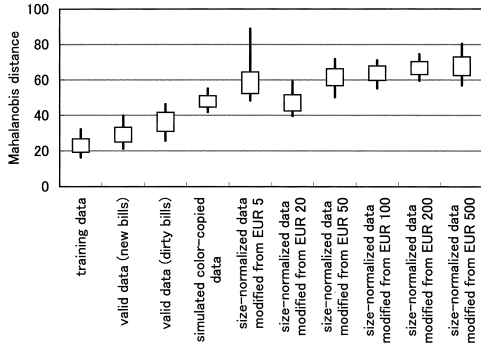**Fig. 7** Mahalanobis distances from EUR 5 image data to training data.

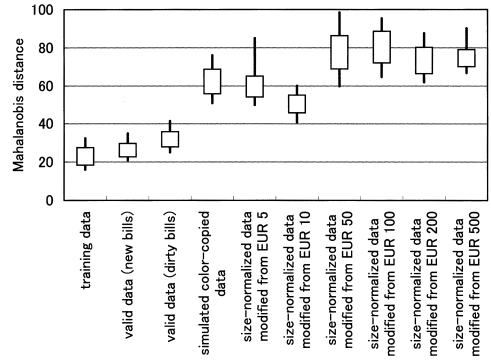**Fig. 8** Mahalanobis distances from EUR 10 image data to training data.

Fig. 9 Mahalanobis distances from EUR 20 image data to training data.
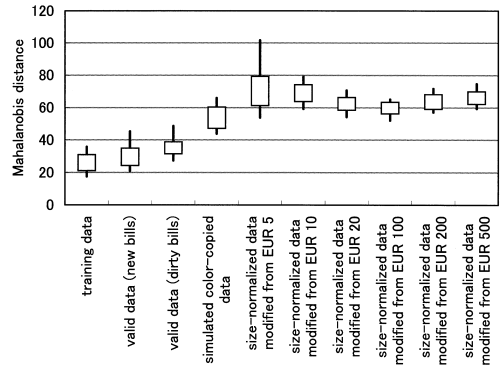

Fig. 10 Mahalanobis distances from EUR 50 image data to training data.

on the size of the EUR 5 (62 [mm]×120 [mm]) because EUR 5 is the smallest one among other Euro banknotes. The sizes of dots are defined as 4 [mm]×4 [mm], 8 [mm]×8 [mm], 16 [mm]×16 [mm] and 32 [mm]×32 [mm]. The number of the hidden neurons in the three-layered perceptron is 64. The initial number of the input neurons is 416 (208 neurons for a visible image and 208 neurons for an IR image), and it is finally reduced to 64.

Each size of available area for the validation part is defined for each banknote and is divided into eight small areas, therefore each validation block has eight RBF networks for each banknote class. Thus, the total number of RBF networks is 8(areas) × 7(classes) = 56. Each size of all dots in image data is 4 [mm]×4 [mm]. The number of the hidden neurons in each RBF network is 20. The number of the input neurons in each RBF network is 60 (for EUR 5 and 10), 72 (for EUR 20 and 50) and 98 (for EUR 100, 200 and 500), respectively.

6. Experimental Results

We have tested several types of configurations and various kinds of input data to verify the performance of our system. This section is divided into three subsections for each experiment. In Section 6.1, we use valid banknotes as input data to the system in order to verify the acceptance performance. In Section 6.2, we use various invalid data as input data to the system in order to verify the rejection performance. In Section 6.3, the system is tested on the various conditions in order to represent superiority of our system.

Before showing the experimental results, we show similarity between training data and experimental data. Mahalanobis distance is a useful way of measuring the similarity between a datum and a data set. The Mahalanobis distance between an experimental datum $\mathbf{x}$ and a training data set S_k for class k is calculated by

$$d^2 = (\mathbf{x} - \mathbf{m}_{S_k})^T \mathbf{C}_{S_k}^{-1} (\mathbf{x} - \mathbf{m}_{S_k}) \qquad (16)$$

where $\mathbf{m}_{S_k}$ is the mean vector of S_k and $\mathbf{C}_{S_k}$ is the covariance matrix for S_k. We used 4 [mm]×4 [mm] resolutional image data obtained by the pre-processing component. **Figures 7–13** show the Mahalanobis distances of training and experimental data from the train-
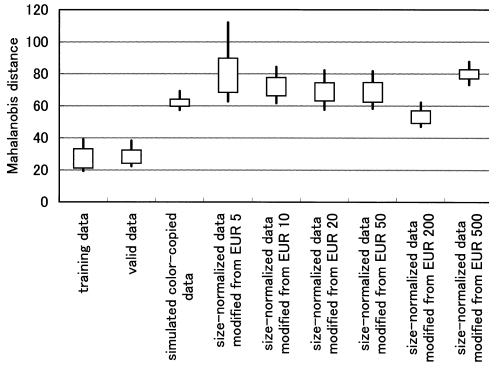
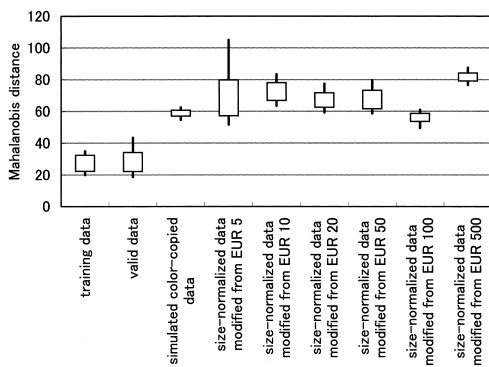**Fig. 11** Mahalanobis distances from EUR 100 image data to training data.

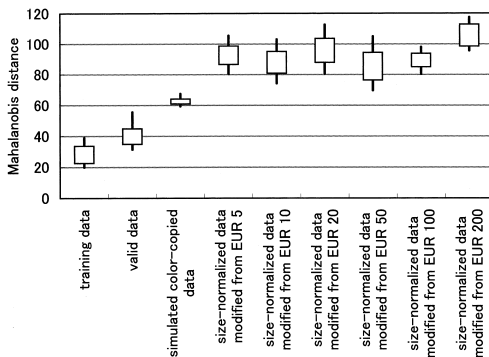**Fig. 12** Mahalanobis distances from EUR 200 image data to training data.

Fig. 13 Mahalanobis distances from EUR 500 image data to training data.

ing data set. The boxes in those diagrams represent the range of the distribution from 10[%] to 90[%].

6.1 Acceptance Performance

We use other 200 pieces of "new" banknotes and 200 pieces of "dirty" banknotes for each kind of Euro banknote to verify the acceptance performance of the system for valid banknotes. **Table 1** and **Table 2** show the acceptance performance of the system. The "classification part" row shows the acceptance rates of the classification part. The "validation part" row shows the acceptance rates of the validation part for classified banknotes. The final acceptance rates of the system are the products of the acceptance rates of the classification part and that of the validation part. They are shown in the "system performance" row.

Table 1 presents the results for the "new" banknotes. Table 2 presents the results for the "dirty" banknotes.

6.2 Rejection Performance

In order to verify the rejection performance of the system for invalid data, we use two sets of the invalid data as the input data to the validation part; simulated color-copied data and size-normailzed data. We have tested the classification part at the same condition in the next subsection as comparative experiments.

Although we should verify the rejection performance for color-copied banknotes, copying banknotes is illegal. We create image data whose properties are close to those of real color-copied banknotes by modifing IR image part of valid data. **Table 3** and **Table 4** show the rejection performance of the validation part for 200 simulated color-copied data for each kind of Euro banknote. Table 3 presents the results for the simulated color-copied data modified from new banknotes. Table 4 presents the results for the simulated color-copied data modified from "dirty" banknotes of EUR 5, 10, 20 and 50.

Next we use size-normalized different kinds of Euro banknotes since the basic design of Euro banknotes resemble another kind of Euro banknotes. **Table 5** and **Table 6** show the rejection performance of the validation part for 200 size-normalized data for each kind of Euro banknote. Table 5 presents the results for the size-normalized data modified from new banknotes. Table 6 presents the results for the size-normalized data modified from "dirty" banknotes of EUR 5, 10, 20 and 50.

Note that the results in Table 3 to Table 6 are indicated by "acceptance rates" for invalid data. If all input data are rejected, the results in those tables become 0.0[%].

6.3 Comparative Experiments

In order to test if our system has a good performance, the system is tested on various conditions.

At first, we verify the rejection performance of the classification part in order to show the

Table 1   Acceptance rates for valid banknotes [%].

	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
classification part	100.0	100.0	100.0	100.0	100.0	100.0	100.0
validation part	100.0	100.0	99.5	100.0	100.0	100.0	100.0
system performance	100.0	100.0	99.5	100.0	100.0	100.0	100.0

Table 2   Acceptance rates for "dirty" banknotes [%].

	EUR 5	EUR 10	EUR 20	EUR 50
classification part	100.0	100.0	100.0	100.0
validation part	99.0	100.0	99.0	100.0
system performance	99.0	100.0	99.0	100.0

Table 3   Acceptance rates for simulated color-copied data in validation part [%].

	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
validation part	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Table 4   Acceptance rates for simulated color-copied data modified from "dirty" banknotes in validation part [%].

	EUR 5	EUR 10	EUR 20	EUR 50
validation part	0.0	0.0	0.0	0.0

Table 5   Acceptance rates for size normalized data in validation part [%].

validation block for	size normalized data of						
	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
EUR 5	–	0.0	0.0	0.0	0.0	0.0	0.0
EUR 10	0.0	–	0.0	0.0	0.0	0.0	0.0
EUR 20	0.0	0.0	–	0.0	0.0	0.0	0.0
EUR 50	0.0	0.0	0.0	–	0.0	0.0	0.0
EUR 100	0.0	0.0	0.0	0.0	–	0.0	0.0
EUR 200	0.0	0.0	0.0	0.0	0.0	–	0.0
EUR 500	0.0	0.0	0.0	0.0	0.0	0.0	–

Table 6   Acceptance rates for size normalized data modified from "dirty" banknotes in validation part [%].

validation block for	size normalized data of			
	EUR 5	EUR 10	EUR 20	EUR 50
EUR 5	–	0.0	0.0	0.0
EUR 10	0.0	–	0.0	0.0
EUR 20	0.0	0.0	–	0.0
EUR 50	0.0	0.0	0.0	–
EUR 100	0.0	0.0	0.0	0.0
EUR 200	0.0	0.0	0.0	0.0
EUR 500	0.0	0.0	0.0	0.0

Table 7   Acceptance rates for simulated color-copied data in classification part [%].

	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
classification part	0.0	31.0	67.5	100.0	44.5	0.0	0.0

necessity of the validation part. **Table 7** shows the rejection performance for simulated color-copied data and **Table 8** shows the rejection performance for size normalized data in the classification part.

Next, we test the validation part by using only visible images as input data in order to appear that IR images are necessary for Euro banknote validation. The acceptance performance for valid banknotes and the rejection performance for size-normalized data are shown in **Table 9** and **Table 10** respectively.

The invalid data used in the experiments for Table 7 and Table 8 are the same data as in Section 6.2. Here, we use 200 invalid data modified from "new" banknotes for each banknote. The valid data used in the experiment for Table 9 are the same data as in Section 6.1. The in-

Table 8   Acceptance rates for size normalized data in classification part [%].

size-based banknote	size normalized data of						
	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
EUR 5	–	98.0	59.0	6.0	3.0	6.5	21.0
EUR 10	97.5	–	100.0	79.0	19.5	19.0	33.0
EUR 20	54.0	99.0	–	99.0	74.5	48.5	15.0
EUR 50	48.5	73.0	99.5	–	100.0	79.5	35.5
EUR 100	0.0	34.0	96.5	100.0	–	100.0	100.0
EUR 200	69.0	61.0	93.5	99.5	100.0	–	100.0
EUR 500	89.0	99.0	99.5	100.0	100.0	100.0	–

Table 9   Acceptance rates for simulated color-copied data in validation part
using only visible image data [%].

	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
validation part	94.0	99.0	99.0	98.0	99.0	94.5	100.0

Table 10   Acceptance rates for size normalized data in validation part
using only visible image data [%].

validation block for	size normalized data of						
	EUR 5	EUR 10	EUR 20	EUR 50	EUR 100	EUR 200	EUR 500
EUR 5	–	11.5	16.0	2.5	0.5	17.0	1.5
EUR 10	11.5	–	14.5	0.0	0.0	0.0	0.0
EUR 20	22.0	22.5	–	0.0	0.0	13.5	0.0
EUR 50	5.0	0.0	0.0	–	0.0	0.5	1.0
EUR 100	4.0	0.0	0.0	22.5	–	34.5	0.0
EUR 200	26.5	0.0	1.5	1.0	0.0	–	0.0
EUR 500	0.0	0.0	0.0	0.0	0.0	0.0	–

valid data used in the experiment for Table 10 are size-normalized data without IR features. Note that the results in all tables are indicated by "acceptance rates". If all input data for rejection test (Table 7, Table 8 and Table 10) are rejected, the results in those tables become 0.0[%].

7.   Discussion

The results of acceptance rates of the system for valid banknotes are shown in Table 1 and Table 2. These results assure that the system has a good performance for accepting valid banknotes. On the other hand, Table 3 to Table 6 show that the results of rejection performance for invalid data are 0.0[%] as acceptance rates. These results show that the validation part performs perfectly for rejecting invalid data in our test. Note that any negative data had not been given to the validation part at learning procedure, that is, RBF networks made boundaries to reject invalid data by themselves. From these results of the acceptance performance and the rejection performance, the system we proposed has a good performance for both accepting valid banknotes and rejecting invalid data.

The results in Table 7 and Table 8 show the necessity of the validation part, i.e., the RBF

networks, in our system. If the rejection performance of the three-layered perceptron is perfect, the RBF networks are not necessary for validation. Table 7 and Table 8 show the classification part in itself has a poor performance for rejecting color-copied data and size-normalized data. The poor performance is based on the fact that the backpropagation method for a three-layered perceptron does not promise performance for unknown data. Thus the classification part should only classify the input data and the validation part is quite important in our system.

The results in Table 9 and Table 10 show the significance of the features in IR images for validation. In these experiments, we verified the performance of the validation part using visible images for accepting valid banknotes and rejecting size-normalized data. The results show that the validation part using only visible images has a poor performrnce for accepting valid banknotes and rejecting invalid data. We have adjusted the output thresholds of the RBF networks in order to improve the acceptance or rejection performance, and it was not able to improve both acceptance and rejection performance at the same time. When the thresholds become higher, the rejection performance

is improved and the acceptance performance becomes worse. When the threshold becomes lower, the acceptance performance is improved and the rejection performance becomes worse. The validation part using only visible images is not tested for rejection performance of color-copied data, since it is obvious that the rejection performance would become worse because that color-copied visible images are almost same as the visible images of valid banknotes. The results show that the IR image in the validation part for Euro banknote is necessary.

Here we discuss the advantages of dividing the system into two parts. We have already mentioned that a three-layered perceptron is not suitable for validation. An RBF network has a function of a classifier and a validator. It is able to configurate the system employing only one RBF network which has as many output neurons as the number of classes. However, it has two problems. First, definition of the available area in image data is complicated. For one RBF network, the size of the available area is fixed for all inserted banknotes, and each banknote class has its own size which is different from other classes. If the size of the available area is fixed to that of the smallest banknote, some parts of image data in a larger banknote are lost. If the size of the available area is fixed to that of the largest banknote, redundant data around a smaller banknote are used as input data. Especially, the size of Euro banknotes has many size variations. Second, the calculation cost increases when the number of the given classes becomes larger. The calculation cost is in proportion to $O(mn^2)$ where m is the number of kernels for each class and n is the number of given classes. The network should have mn hidden neurons because the accuracy of an RBF network is determined by the number of the hidden neurons assigned to each classes. Therefore, for n output neurons, the number of connections between the hidden layer and the output layer is $mn \times n = mn^2$. Our proposed system, which is divided into two parts, solves these problems. For the first problem, each size of the available area can be defined simply for each class because the validation part has each validation block for each banknote class. The system can use image data efficiently for validation. For the second problem, the size of each RBF network does not depend on the number of the classes because

of the independence of the validation part from the classification part. Each RBF network has m hidden neurons respectively and the calculation cost of the validation part is in propotion to $O(m)$. The calculation cost does not increase when the number of the classes increases. While the size of the three-layered perceptron in the classification part depends on the number of classes, the classification part is less influenced than the validation part because the size of the three-layered percetron becomes smaller than the size of the RBF networks by reducing redundant input neurons.

8. Conclusion

We propose an Euro banknote recognitnion system composed of a three-layered perceptron for classification and RBF networks for validation. We also presented to use IR and visible images as input data to the system.

The system has a good performance for both accepting valid banknotes and rejecting invalid data. By comparing experimental results, we showed the classification part using a three-layered perceptron cannot reject all invalid data in itself, and it is effective to use the validation part using RBF networks. In addition, the advantages in dividing the system into two parts are shown in our discussion. We also verify the performance of the validation part without IR images, and the results show that IR images are quite significant for Euro banknote validation.

References

1) Katoh, M., Ohmae, K., Sonoda, S., Yanagida, M. and Senga, M.: Image processing device and method for identifying an input image, and copier scanner and printer including same, United States Patent 5-845-008 (1998).
2) Fan, Z., Wu, J.W., Micco, F.A., Chen, M.C. and Phong, K.A.: Method for counterfeit currency detection using orthogonal line comparison, United States Patent 6-181-813 (2001).
3) Takeda, F., Omatu, S., Inoue, T. and Onami, S.: High Speed Conveyed Bill Money Recognition with Neural Network, *Proc. IMACS/SCINE International Symposium on Robotics, Mechatronics and Manufacturing Systems*, '92 Kobe, Japan, Vol.1, pp.16–20 (1992).
4) Takeda, F. and Omatu, S.: A Neuro-Money Recognition Using Optimized Masks by GA,

Advances in Fuzzy Logic, Neural Networks and Genetic Algorithms LNAI 1011, pp.190–201, Springer (1995).

5) Kosaka, T. and Omatu, S.: Classification of the Italian Liras Using the LVQ Method, *International Joint Conference on Neural Networks 2000*, Vol.3, pp.143–148 (2000).

6) Takeda, F., Tanaka, M., Nishikage, T. and Fujita, Y.: A Rejecting Method of Non-objective Currencies using Multi-Dimensional Gaussian Probability Density Function for Neuro-Multi National Currency Recognition, *T. IEE Japan*, Vol.118-D, No.12, pp.1361–1369 (1998).

7) Eccles, N.J.: Neural network for banknote recognition and authentication, United States Patent 5-619-620 (1997).

8) Baudat, G.: Method and apparatus for the classification of an article, United States Patent 5-678-677 (1997).

9) Broomhead, D. and Lowe, D.: Multivariable functional interpolation and adaptive networks, *Complex Systems*, Vol.2, pp.321–355 (1988).

10) Sato, K., Takahashi, Y. and Nagasawa, Y.: A Study on Partial Face Recognition using Radial Basis Function Networks, Technical Report of IEICE, PRMU97-163, pp.83–89 (1997).

11) Liu, J. and Gader, P.: Outlier Rejection with MLPs and Variants of RBF Networks, *Proc. International Conference on Pattern Recognition (ICPR'00)*, pp.684–687 (2000).

12) Burr, D.J.: Experiments with a connectionist text reader, *Proc. IEEE 1st International Conference on Neural Networks*, Vol.4, pp.717–724 (1987).

13) LeCun, Y.: Backpropagation applied to handwritten zip code, *Neural Computation*, Vol.1, pp.541–551 (1989).

14) Obaidat, M.S. and Abu-Saymeh, D.S.: Performance Comparison of Neural Networks and Pattern Recognition Techniques for Classifying, *Proc. 1992 ACM/SIGAPP symposium on Applied computing: technological challenges of the 1990's*, pp.1234–1242 (1992).

15) Bellman, R.E.: *Adaptive Control Processes*, Princeton University Press (1961).

16) Matsunaga, Y., Murase, K., Yamakawa, O. and Tanifuji, M.: A Modified Back-Propagation Algorithm that Automatically Removes Redundant Hidden Units by Competition, *IEICE (D-II)*, Vol.J79-D-II, No.3, pp.403–412 (1996).

17) Moody, J. and Darken, C.J.: Fast learning in networks of locally tunes processing units, *Neural Computation*, Vol.1, pp.284–294 (1989).

18) Hertz, J., Krogh, A. and Palmer, R.G.: *Introduction to the theory of neural computation*, New York, Addison Wesley (1991).

19) Schwenker, F., Kestler, H.A. and Palm, G.: Three learning phases for radial-basis-function networks, *Neural Networks*, Vol.14, pp.439–458 (2001).

20) Kohonen, T.: *Self-Organizing Maps*, Springer-Verlag, Berlin (1995).

21) Rumelhart, D.E., McClelland, J.L. and PDP Research Group: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol.1, MIT Press, Cambridge (1986). (distributed to authors).

Masato Aoba is a Ph.D. student at Keio University since 2002. He is also a researcher from TOYO Communication Equipment Co., Ltd. (TOYOCOM). He received his MS degree in Mechanical Engineering from Waseda University in 1995. Since 1995, he worked at TOYOCOM. His research interests focus on neural networks and their application.

Tetsuo Kikuchi works at TOYOCOM since 1995, developing banknote recognition units. He received his BS degree in Electrical Engineering from Tokyo University of Science in 1995. His research interests focus on neural computing and hyperspectral computing.

Yoshiyasu Takefuji is a tenured professor on faculty of environmental information at Keio University since April 1992 and was on tenured faculty of Electrical Engineering at Case Western Reserve University since 1988. Before joining Case, he taught at the University of South Florida and the University of South Carolina. He received his BS (1978), MS (1980), and Ph.D. (1983) in Electrical Engineering from Keio University. His research interests focus on neural computing and hyperspectral computing.