Paper

A Reliable Classification Method for Paper Currency Based on the Non-Linear PCA

Ali Ahmadi[*] Member
Sigeru Omatu[*] Member
Toshihisa Kosaka[**] Member

This paper addresses the reliability of neuro-classifiers for paper currency recognition. A local principal component analysis (PCA) method is applied to remove non-linear dependencies among variables and extract the main principal features of data. At first the data space is partitioned into regions by using a self-organizing map (SOM) clustering and then the PCA is performed on each region. A learning vector quantization (LVQ) network is employed as the main classifier of the system. By defining a new algorithm for rating the reliability and using a set of test data, we estimate the reliability of the system. The experimental results taken from 1,200 samples of US dollar bills show that the reliability is increased up to 100% when the number of regions as well as number of codebook vectors in the LVQ classifier are taken properly.

Keywords : local PCA, reliability, LVQ, SOM, paper currency recognition.

1. Introduction

Recently, the recognition of paper currency has been concerned effectively by using neural networks[(1) (2)], and it is shown that neuro-classifiers are robust for recognition of defective, taint, and worn out paper currency. Takeda et al[(3)]. have used a random mask for preprocessing the data and then a multi-layer neural network as the classifier for recognition of paper currency. Teranishi et al[(4)]. have applied a method based on acoustic cepstrum patterns for extracting the features of bill and then a competitive neural network as the classifier. Tanaka[(5)] has employed a probabilistic principal component analysis (PCA) for extracting the main characteristics of bill data.

Due to high risk of misclassification in such systems, the reliability of recognition becomes of high importance. Basically the classifier must be fully robust for frayed or dirty bills of different models, and also has insensitivity to shift rotation and different directions of inserting bill. In fact, even if the classification rate is 100% over the data space, still it is necessary to make sure about the reliability of classification over all variety of real data.

We have already proposed in our pervious study[(6)] a method which uses the PCA algorithm for extracting the features of bill image data and utilizes a learning vector quantization (LVQ) network as the main classifier, where the reliability is evaluated thorough a new defined algorithm using Gaussian mixture densities for distribution of data. We have found that in case of large variability in input data or any non-linear correlation among the data, some additional discrimination process is needed to keep the reliability high enough.

As the main limitation of the PCA is its global linearity, that is, it only defines a linear projection of data and does not model non-linear relationship among variables, some developments of non-linear principal component analysis (NLPCA) have been presented to address this limitation[(7)]. However, both PCA and NLPCA algorithms try to model the entire data by the same global features. As an alternative, the complexity of the data can be modeled by using a mixture of the local linear PCA. The local PCA algorithm clusters the input data into regions and performs PCA on the data that falls within each region.

In this paper, we apply a local PCA method where a self-organizing map (SOM) is used for clustering the data into homogenous regions. Our approach is similar to Kerschen et el[(8)] in the sense of local PCA application but differs in the clustering method as they have used vector quantization (VQ) for the clustering phase. This will be explained in more details in following Chapters.

The current system is intended for classifying different kinds of paper currency, however, we examined only US dollar bills. The experimental results show a growth in reliability by 3.5% after using features extracted by the local PCA method comparing with the method based on the conventional PCA and a increment by 7.2% comparing with classification without the PCA.

2. An Outlook of the Proposed System

Figure 1 shows a block diagram of the system proposed in this paper. After preprocessing the data a SOM is used to cluster the data into a map of regions. Then by using the local PCA on each region, a set of 30 dimensional vectors is extracted as the main features of bill data. Next, an LVQ network is used as the main classifier of the system which classifies feature vectors of data to 24 output classes, which means 6 kinds of US dollar bills and 4 directions for each bill. The reliability of classification is evaluated by using a newly defined algorithm which applies the probability density functions of test data. To reach to a 100% reliability rate we keep modifying of system parameters including

[*] Department of Computer and System Sciences, Osaka Prefecture University
Sakai, Osaka 599-8531
[**] Glory Ltd.
Himeji, Hyogo 670-8567

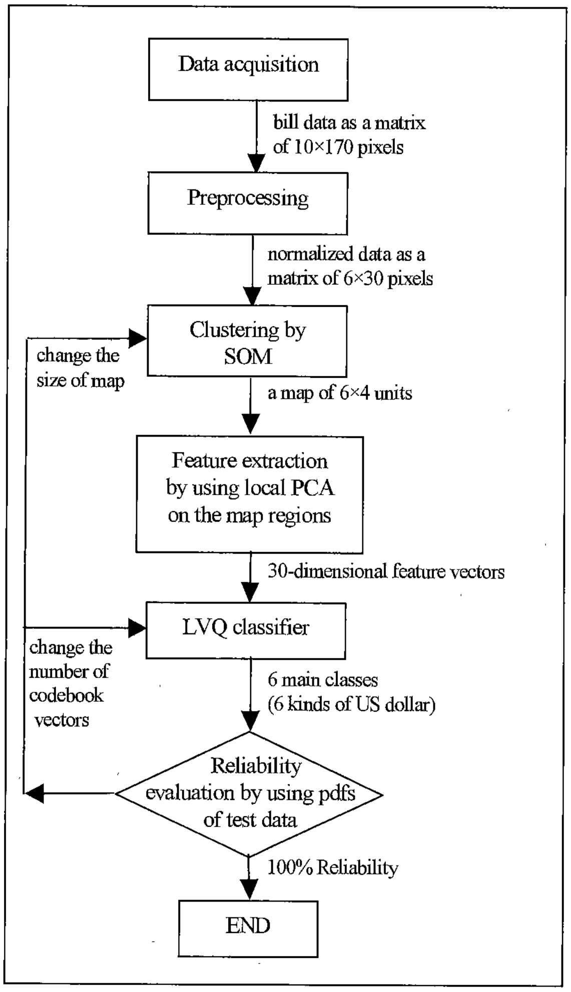

Data acquisition
bill data as a matrix of 10×170 pixels
Preprocessing
normalized data as a matrix of 6×30 pixels
Clustering by SOM
change the size of map
a map of 6×4 units
Feature extraction by using local PCA on the map regions
30-dimensional feature vectors
LVQ classifier
change the number of codebook vectors
6 main classes (6 kinds of US dollar)
Reliability evaluation by using pdfs of test data
100% Reliability
END

Fig. 1. An outlook of the paper currency recognition system.

the number of regions for the local PCA as well as the number of codebook vectors for the LVQ classifier.

3. Preprocessing Data

The original image of bill money is acquired through five advanced sensors. Each sensor uses two different wave lengths for generating two channels of data. Thus, there are a total of 10 channels each of them contains 170 pixels (i.e., an 10×170 array). At first by using a linear function we generate a new channel of data based on the data of two channels for each sensor. Hence, 15 channels are obtained in total among which we select 6 main channels that represent the main characteristic of data. Theses channels are the central located channels with less similar pixels. In order to compress the 170 data pixels in each channel we discard the first and last 10 pixels and take the average of every five neighbor pixels so that finally we have 30 pixels in each channel. Then a linear transformation is applied for normalization data as follows:

$$x_i = \frac{t_i - \bar{t}}{S_t} G + C \quad \cdots\cdots (1)$$

where t_i is the pixel value in each channel, $\bar{t}$ is the mean value of pixels, S_t is the standard deviation, and G=512 and C=128 are the coefficients of gain and offset, respectively whose values are determined experimentally. Thus, a vector $\mathbf{x}$ of 6×30(=180) elements is provided for using in the feature extraction step.

4. PCA Feature Extraction

PCA is one of the most popular methods for preprocessing, compression, and feature extraction of data and it is discussed in most documents on multivariate analysis [(9)]. The most common derivation of the PCA is in terms of a linear projection which maximizes the variance in the projected space. As explained in the introduction, the PCA only removes linear correlation among the data and is only sensitive to second order statistics, i.e., it is assumed that the distribution of data is Gaussian. In case of non-linear relations among the variables, we need to consider higher order statistics to eliminate the dependencies which are not removed by the PCA. Here, we apply a local PCA model where the data is clustered into regions by using a SOM clustering at first and then the PCA is performed on the data of each region. The procedure is explained in the following.

4.1 SOM Clustering SOM is shown to have desirable properties compared to classical clustering methods. It provides a natural measure for the Euclidean distance of a vector from a cluster which is adaptive from the local statistics of the data[(10)]. The SOM forms a map corresponding to the data distribution so that regions of the map can be interpreted as clusters in the data space. The main key point is to define a set of codebook vectors $\mathbf{m}_i$, i=1,2,..,q which represent units of the map. Then as for a given input $\mathbf{x}$, it is mapped to a unit associated with $\mathbf{m}_c$ such that

$$\| \mathbf{x} - \mathbf{m}_c \| = \{\| \mathbf{x} - \mathbf{m}_i \|\}. \quad \cdots\cdots (2)$$

Then $\mathbf{m}_i$ codebook vectors are updated through a training process iteratively as

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h_{ci}(t)\,[\mathbf{x}(t) - \mathbf{m}_i(t)] \quad \cdots\cdots (3)$$

where t indicates the iteration and $hci(t)$ is a neighborhood function taken as

$$h_{ci}(t) = \alpha(t) \exp(-d_{ci}^2/2\, r^2(t)). \quad \cdots\cdots (4)$$

Here, $\alpha(t)$ and $r(t)$ are learning rate and neighborhood radius, respectively, both decrease monotonically as a linear function of time, and d_{ci} is the distance between mc and $\mathbf{m}_i$. The parameter values used for the SOM clustering are shown in Table 1. As it can be seen a 6×4 map is used for clustering the preprocessed data into 24 regions.

Table 1. Parameters values used in SOM clustering.

Parameter	Value
Map size	6 x 4
Topology	Hexagonal
Learning rate function	Linear
Neighborhood function	Gaussian
Initial learning rate in first phase	0.2
Initial learning rate in second phase	0.05
Initial neighborhood radius in first phase	10
Initial neighborhood radius in second phase	4

Comparing with the other usual VQ clustering methods(8)(11) SOM has the advantage of using neighborhood function for retrieving the codebook vectors which gives the potentiality of smoothing classifier parameters without need to more new training data. Moreover, the SOM is beneficial in the sense of creating a two dimensional map whose units can be easily addressed in the posterior stages of classification.

4.2 Local PCA Modeling If $\mathbf{x}_i$ is supposed to be an m-dimensional vector of a dataset with i=1,...,N, then the goal of the PCA is to find r dimensional axes $\mathbf{p}_i$ onto which the retained variance under projection is maximal(13). These axes are given by the eigenvectors associated with r largest eigenvalues of the covariance matrix of data as:

$$\Sigma\,\Phi=\Lambda\,\Phi \quad\cdots\cdots\cdots\cdots\cdots (5)$$

where Σ is the covariance matrix of data $\mathbf{x}$, $\Phi = [\mathbf{p}_1,\mathbf{p}_2,\ldots,\mathbf{p}_m]$ is the eigenvectors matrix, and Λ= diag$\{\lambda_1,\lambda_2,\ldots,\lambda_\mu\}$ is the eigenvalue matrix with $\lambda_1>\lambda_2>\ldots>\lambda_m$. Then the transformed data vector $\mathbf{y}_i$ is determined as

$$\mathbf{y}_i = f(\mathbf{x}_i) = \ \Phi^{\mathrm{T}}\ (\mathbf{x}_i\ -\ \mu) \quad\cdots\cdots\cdots\cdots\cdots (6)$$

which is a reduced r-dimensional representation of data vector $\mathbf{x}_i$ with m dimensions.

Here, considering the q local regions which have been already provided by the SOM, we apply a sort of functions f_j (.) with j=1,...,q instead of a single encoding function f. The procedure is taken place as follows: For each cluster of data corresponding to each region of the SOM with the mean vector of μ_j, the covariance matrix is estimated as:

$$\Sigma_j = \frac{1}{N_j}\sum_{\mathbf{x}\in S_j}(\mathbf{x}-\mu_j)(\mathbf{x}-\mu_j)^T \quad\cdots\cdots\cdots\cdots\cdots (7)$$

where N_j is the number of vectors lied in the cluster S_j. Then by determining the eigenvectors ($\mathbf{p}_{j1}$, ..., $\mathbf{p}_{jr}$) of each matrix Σ_j, the function f_j and thereby the transformed vector $\mathbf{y}_i$ can be obtained for each region as follows:

$$\mathbf{y}_i = f_j(\mathbf{x}_i) = [\mathbf{p}_{j1},\ \ldots,\ \mathbf{p}_{jr}]^{\mathrm{T}}\ (\mathbf{x}_i - \mu_j)\ , \qquad \mathbf{x}_i \in S_j\ . \cdots\cdots\cdots (8)$$

As explained in Chapter 3, in this paper the data dimension m is 180 and r is taken as 30. The number of regions q is taken as 24 according to the SOM clusters as described in Section 4.1.

Accordingly, through application of the PCA over all regions a new 30 dimensional dataset is produced which contains the main features of 180 dimensional data.

5. LVQ Classifier

Kohonen's LVQ is a supervised learning algorithm associated with the competitive network(10) which basically consists of an input layer and an output layer, and an array of weight vectors $\mathbf{w}_i$ = $[\mathbf{w}_{i1},\ \mathbf{w}_{i2},\ \ldots,\ \mathbf{w}_{in}]^{\mathrm{T}}$ where w_{ij} denotes a connection weight between the j-th node in the input layer and i-th node in the output layer (Fig. 2). Given a training dataset X, each labeled with a class identifier, and a set M of codebook vectors, the LVQ network adaptively modifies these codebooks so that they represent the class probability distribution in the training dataset. This modification of codebooks consists of applying a "punishment" when a codebook is near a sample of a different class and "reward" when it is near a sample of its own class.

Since the LVQ network is beneficial in classification of data with a large number of inputs and explanation of the misclassification, it is applied as the main classifier of the present system. As we consider 6 kinds of US bills including 1, 5, 10, 20, 50, and 100 dollars and for each bill there exist four directions of rotation (Figs. 3 and 4), a total of 24 (=6×4) output categories are considered for the classifier.

It is worth noting that although the four rotations of a bill are indeed belonged to the same class but as we use fixed sensors for reading data pixels, the acquired data of different directions are

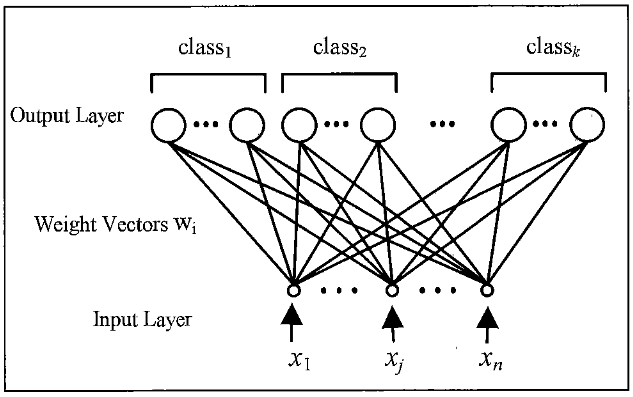


Fig. 2. A schema of LVQ network structure.

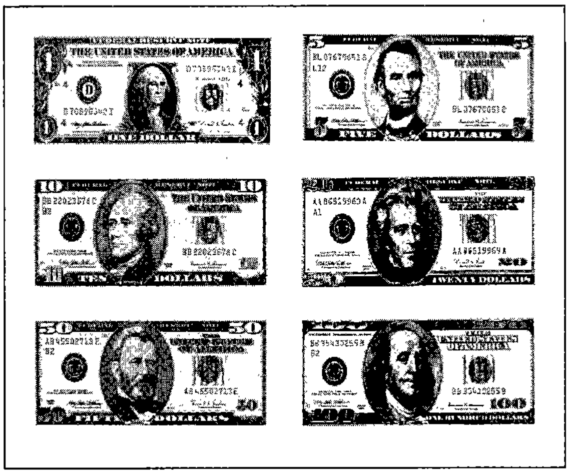


Fig. 3. Six different US dollar bills used here, including 1,5,10,20,50, and 100 dollar.

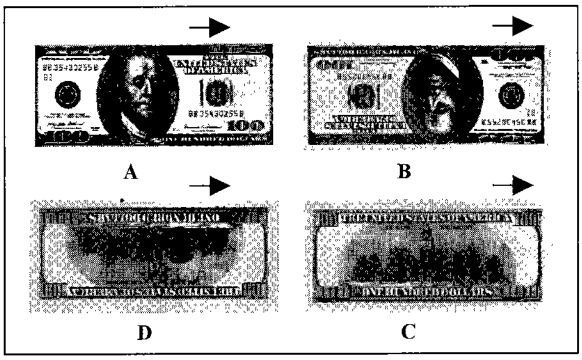


Fig. 4. Four different directions for inserting a bill.

quite different. So we need to distinguish these subclasses in classifier however finally in reliability evaluation we consider them as the same class.

The system is trained by taking a trial number of codebook vectors for each class looking for the best classification rate and maximum reliability. A total number of 120 codebook vectors (averagely 5 vectors per class) is experimentally found to be the best. The number of iterations for each training epoch is taken to be 10,000 while a linear function as $\alpha(t) = \alpha(0)\,(1.0 - t\,/\,T)$ is applied for learning where T is the number of iterations. Therefore, the LVQ classifier has 30 neurons (the number of extracted features) in the input layer and 120 neurons in the output.

6. Reliability Evaluation

There exist variety of methods for estimating the accuracy of classifiers within which two methods are most common: Cross-validation and Bootstrapping$^{(12)}$. Cross-validation itself has two major types: *K*-fold cross-validation and leave-one-out. In *k*-fold cross-validation the dataset is randomly divided into *k* mutually exclusive subsets (the folds) of approximately equal size. The system is trained and tested *k* times, each time tested on a fold and trained on the dataset minus the fold. The cross-validation estimate of accuracy is the average of the estimated accuracies from *k* folds. If *k* equals the sample size, this is called leave-one-out cross-validation.

Bootstrapping is an improvement on cross-validation method. In the simplest form of bootstrapping, instead of repeatedly analyzing subsets of the data, we repeatedly analyze sub-samples of the data. Each sub-sample is a random sample with replacement from the full sample. In the .632 bootstrap which is the popular one, in a given dataset of size *n*, a bootstrap sample is created by sampling *n* instances uniformly from the data (with replacement). The accuracy estimate is derived by using the bootstrap sample for training and the rest of the instances for testing.

As it can be realized both of cross-validation and bootstrapping methods are based on re-sampling the test data for estimating the accuracy of the classifier. But in this paper we are concentrating on reliability evaluation rather than accuracy estimation. In fact in real cases even a well-trained network (i.e., a net which has reached a low error value at the end of the training phase) can provide output vectors considerably different from what are expected; this is mainly due to unavoidable distortions which affect the real data samples, making them quite different from the ones in the training set. It is thus necessary to identify a reliability criterion which is less dependent on the test data and can evaluate the reliability of the system when it concerns with the real data space. So, we propose a simple but effective algorithm which is well-adjusted to the LVQ classifier and is based on determining the probability densities of data around codebook vectors and then calculating the overlaps between densities of different classes. Here, for simplicity the distribution of data is supposed to be normal (Gaussian). The process is as follows: According to simple-split method, the dataset is split into two disjoint sets of instances: training set and test set. After the classifier is trained by the training set and the codebook vectors are determined, the test data set is used to estimate the parameters of the probability density function (pdf) around each codebook vector as

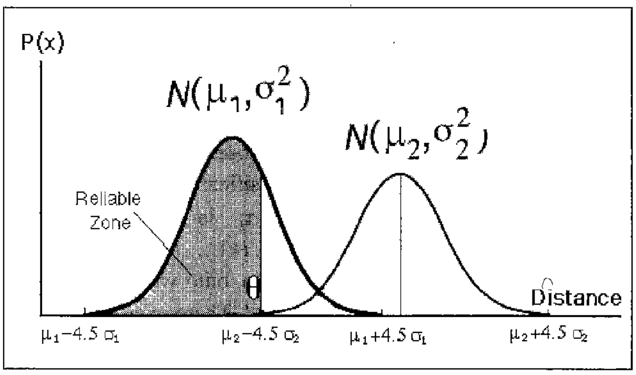


Fig. 5. The overlap between codebook densities of two nearby classes.

$$p_i(\xi) = 1/(\sqrt{2\pi}\,\sigma_i)\ \exp\left(-(\xi-\mu_i)^2\,/\,2\sigma_i^{\,2}\right) \quad \cdots\cdots\cdots\cdots (9)$$

where ξ (=‖ $\mathbf{x}_i$ - $\mathbf{m}_i$ ‖) is the distance between data vector and codebook vector, μ_i and σ_i are the mean and variance in pdf of codebook vector *i*, respectively. Assuming the Gaussian pdf, the interval [$\mu_i - 4.5\sigma_i$, $\mu_i + 4.5\sigma_i$] can be considered as an area that covers almost 100% of probabilities (=100–5.122×10^{-5}). For a given class of codebook vectors if the densities have no overlap within this interval with densities of other classes, the reliability for this class is supposed to be 100%. But in the case that this interval is overlapped with distributions of another classes, the reliability can be calculated as

$$RM = \sum_{i}^{L_k} \alpha_i^k \int_{\mu_i-4.5\sigma_i}^{\theta_i} p_i(\xi)\, d\xi \quad \cdots\cdots\cdots\cdots\cdots\cdots (10)$$

where L_k is the number of codebook vectors within each class *k*, α_i^k is a normalizing coefficient for each class ($\Sigma\alpha_i$=1, α_i> 0), here, it is taken as the number of vectors belong to each codebook vector region divided by the total number of vectors belong to the same class, and θ_i is the cross point of each pdf with the interval boundary of nearest density from another class (Fig.5).

Thus, the total reliability rate of the system finally can be determined by averaging these class reliability values.

Comparing with other accuracy estimation methods mentioned above, the proposed method is adventagous in the sense of less dependence on the test data and giving a more continuous fuction for prediction of the reliability. Moreover, since both the cross-validation and bootstrap algorithms are based on re-sampling the data and it requires the system to be retrained many times, the speed of the training gets down, but in the proposed method the sample-splitting and also training of the system is done only once, so it hightens the speed of the system significantly significatly.

7. Experimental Results and Discussion

A dataset of 3,600 samples from 6 kinds of US dollar bills including 1, 5, 10, 20, 50, and 100 US dollar (taking four directions of rotation for each bill, i.e., a total of 150 samples for each direction) is used for examining the system. The bills we used were of various level of fatigue and issued in different years. However, globally they can be considered as normal bills not frayed ones. Figures 6 shows the distribution of data for different

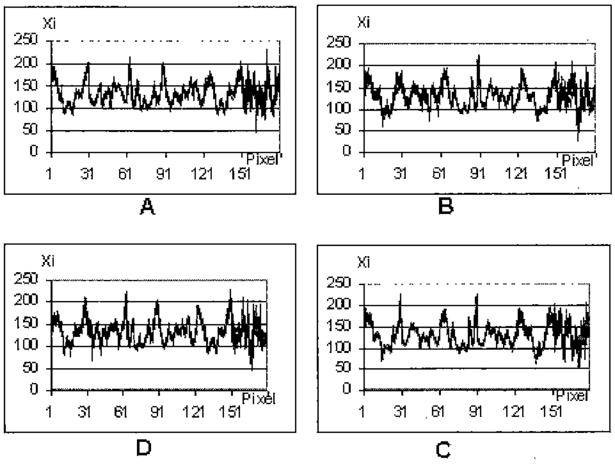


Fig. 6. Distribution of data in four directions of 100 US dollar bill (taken from 30 samples).

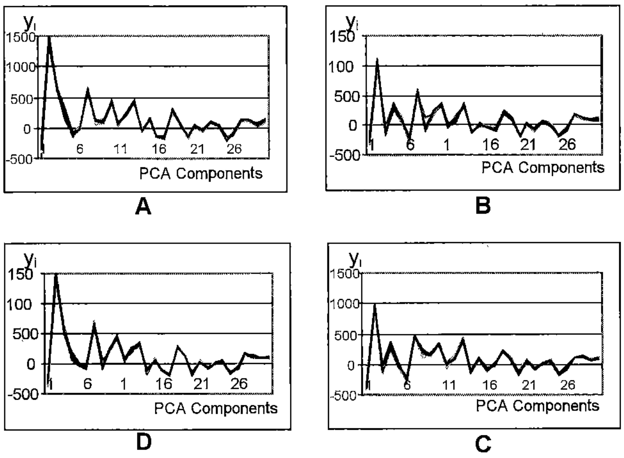


Fig. 7. Distribution of data for 100 US dollar bill by using 30 PCA components (taken from 30 samples).

directions of 100 US dollar bill, and Fig.7 shows the distribution of the same data after extracting main features by PCA.

At first using the split-sample method, the main dataset is divided into two subsets: Training (65%, i.e., around 2,400 samples) and Test (35%, i.e., around 1,200 samples). The classifier is trained using the training set of data and the test set is used for evaluating the classification rate. As for learning the LVQ classifier we have tried different numbers of codebook vectors from 3 to 10 per class looking for the best result of classification.

On the other hand, concerning to the local PCA application, we have tried different numbers of regions from 12 to 30 as the output of the SOM to study its influence on the classification. The results of classification rate are shown in Table 2. As it is shown, in case of 120 or 200 codebook vectors the recognition rate is 100% for all different numbers of regions.

Table 2. The results of classification on test data (split-sample method), by using the local PCA and taking different numbers of regions and codebook vectors.

No. of Codebooks	No. of Regions	Recog. Rate (%)
60	1 (Standard PCA)	81
	12	83.2
	24	86
	30	86
80	1 (Standard PCA)	87 1
	12	91
	24	92
	30	92
120	1 (Standard PCA)	98
	12	100
	24	**100**
	30	100
200	1 (Standard PCA)	99.1
	12	100
	24	100
	30	100

Table 3. The results of classification using *k*-fold cross-validation method taking *k* from 2 to 20. The numbers of regions for local PCA are taken from 1 to 30. The number of codebook vectors is 120.

No of Regions	k=2	k=5	k=10	k=20
1 (Standard PCA)	95.3	96	100	100
12	97.1	97.4	100	100
24	97.5	96	100	100
30	97.5	96.4	100	100

Table 4. The results of classification using bootstrap method taking samples number from 100 to 1,500. The numbers of regions for local PCA are taken from 1 to 30. The number of codebook vectors is 120.

No. of Regions	No. of Samples				
	100	200	500	1000	1500
1 (Standard PCA)	92	97.2	97	100	99 1
12	92.4	100	97.4	100	100
24	93	100	98	100	100
30	93	100	98	100	100

Before using the proposed reliability evaluation method and in order to have comparative results, we apply cross-validation (*k*-fold) and .632 bootstrap algorithms. The results of estimated accuracy for different numbers of *k* (in cross-validation) and different numbers of samples (in bootstrap) are reported in Table 3 and Table 4. As it can be seen in Table 3, for low values of *k* in *k*-fold method the classification rate is down but by taking higher values of *k* and increasing the number of local PCA regions, the classification rate will be increased. The *k*=10 seems to be the best choice concerning the speed and bias of the system.

The same situation conducts for bootstrap results in Table 4 whereas increasing in number of samples causes improvement of accuracy rate, however, this makes low speed of training.

The main results are then obtained by application of the reliability evaluation method proposed in this paper. Again the test dataset (35%) is used and different numbers of codebook vectors as well as different numbers of SOM regions are tried looking for the best reliability rate. The results are shown in Table 5. As it can

Table 5. The results of reliability on test data, by using the local PCA and taking different numbers of regions and codebook vectors.

No. of Codebooks	No. of Regions	Reli. Rate (%)
60	1 (Standard PCA)	75.1
	12	77
	24	80.1
	30	81
80	1 (Standard PCA)	82
	12	84.2
	24	86.3
	30	86.3
120	1 (Standard PCA)	96.5
	12	97.8
	24	**100**
	30	100
200	1 (Standard PCA)	95
	12	97
	24	100
	30	100

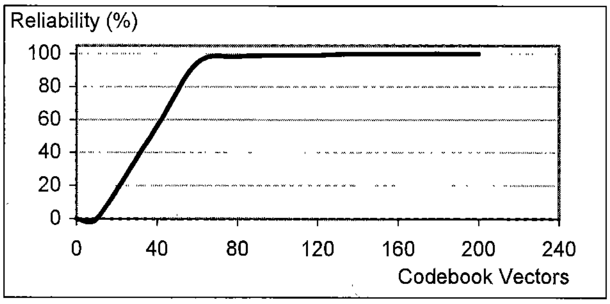


Fig. 8. The relation between reliability and number of codebook vectors.

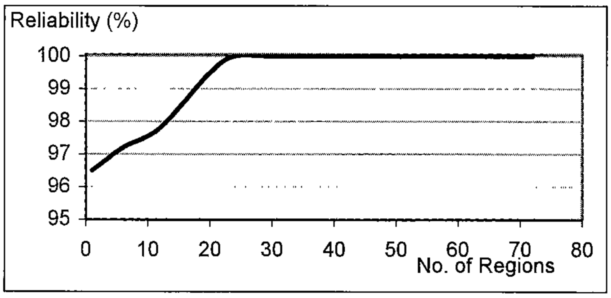


Fig. 9. The relation between reliability and number of regions in local PCA.

be seen by increasing the number of codebook vectors in the classifier as well as the number of regions in the local PCA, the reliability can be increased significantly. The 100% reliability rate can be obtained in both cases of 120 and 200 codebook vectors when the number of regions is taken to be 24 or 30. But due to complexity of the system and considering that the larger number of codebooks in LVQ classifier basically reduces the speed of training and classification, we select the case of 120 codebook vectors. The same situation conducts for number of regions and we found the number 24 to be the best for current experiment.

Figures 8 and 9 indicate the relation between number of codebook vectors and number of regions with reliability of system, clearly. As we can see in Fig. 8, increasing the number of codebooks makes a high increment in reliability value firstly but after a certain point, the reliability is not influenced anymore by the larger number of codebook vectors. Similar situation conducts in the Fig. 9 for relation between the reliability and number of regions, but once the number of regions increases so much, it makes an inverse affect on the reliability rate. As it is displayed, the minimum value of reliability is occurred when the number of regions is one, that is, when we use a conventional PCA instead of the local PCA. It means that we have obtained a significant growth of 3.5% in reliability rate compared to our pervious work which was based on using the standard PCA.

We have also changed the number of PCA components from 15 to 30 to investigate the impact on reliability. The results show that it also has a direct affect on heightening reliability but when the number of components drops lower than a threshold value, the reliability starts to decrease. The reason can be explained in the way that the PCA, however, increases the variance within the new components space, but as the distances between codebooks are increased, it makes the overlap between probability density functions to be significantly declined and consequently the reliability of the system is improved. But when the components are very few there would be a high variance in pdfs which will enlarge the overlap zones and therefore, reduce the reliability. We have found experimentally that a choice of 30 components is the best.

If we have an overview on the results of different methods, we can see that in case of 120 codebook vectors the results of classification are 100% in the first three methods (i.e., simple split-sample, *k*-fold cross-validation, and .632 bootstrap) regardless of the number of regions in the local PCA. But in our proposed method the reliability rate is 100% only when the number of regions is taken to be 24 or more. This indicates that this method is of more precision. Moreover, as the results show, the continuity of this method looks to be more comparing to the other methods.

8. Conclusion

In this paper we have presented a local PCA approach for feature extraction of data in classification of paper currency. The aim is to model the complexity of data and correlation among variables by using a simple linear model. The experimental results taken from 3,600 US dollar bills show that by taking a proper number of regions and also an optimized number of codebook vectors for the LVQ classifier, the reliability of the system can be increased up to 100%. Comparing with the conventional PCA method which was our pervious approach, the present method shows a significant growth in reliability rate.

Also, the method for reliability evaluation proposed in this paper has the advantage of higher speed and less dependence on the test data comparing to other accuracy estimation methods. However, as we still suppose normal distributions for data around codebook vectors, there is some approximation in the results, and the results are still dependent on the test data.

As all the test samples used in this research were supposed to be from known classes (different US dollar bills) and there were no unknown bills within the data, we didn't consider any reject case for classification but due to high risk of misclassification in such systems and in order to make the system still more reliable, we are planning to consider a reject option in our next work. Also, to extend this study to some new boundaries and improve the system for application in much wider variety of data bills, our next work will be exploiting this method with other types of classifiers instead of linear neural network LVQ.

(Manuscript received January 24, 2003, revised May 21, 2003)

References

(1) A. Frosini, M.Gori, and P.Priami : "A Neural Network-based Model for Paper Currency Recognition and Verification", *IEEE Trans. Neural Networks*, **7**, No. 6, pp. 1482-1490 (1996-6)

(2) T. Kosaka, N. Taketani, and S. Omatu : "Classification of Italian Bills by a Competitive Neural Network", *The Trans. of The Institute of Electrical Engineers of Japan*, **119-C**, No. 8/9, pp. 948-954 (1999-8/9)

(3) F. Takeda and S. Omatu : "Bank Note Recognition System Using Neural Network with Random Masks", *Proc. of the World Congress on Neural Networks*, **1**, pp. 241-244, Portland, USA (1993)

(4) M. Teranishi, S. Omatu, and T. Kosaka : "Neuro-classifier of Currency Fatigue Level Based on Acoustic Cepstrum Patterns", *Journal of Advanced Computational Intelligence*, **4**, No. 1, pp. 18-23 (2000-1)

(5) M. Tanaka : "Modeling of Mixtures of Principal Component Analysis Model with Genetic Algorithm", Proc. *31st Int'l Symposium on Stochastic Syst. Theory and Its Appl.*, pp. 157-162, Yokohama, Japan (1999)

(6) A. Ahmadi and S. Omatu : "A High Reliability Method for Classification of Paper Currency Based on Neural Networks", *Proc. of The Eighth International Symposium on Artificial Life and Robotics (AROB 8th '03)*, **2**, pp. 601-604, Oita, Japan (2003)

(7) M.A. Kramer : "Non-linear Principal Components analysis Using Auto-associative Neural Networks", *A.I.Ch.E Journal*, **37**, pp. 233-243 (1991)

(8) G. Kerschen and J.C. Golinval : "Non-linear Generalization of Principal Components Analysis: From A Global to Local Approach", *Journal of Sound and Vibration*, **254**, No. 5, pp. 867-876 (2002-5)

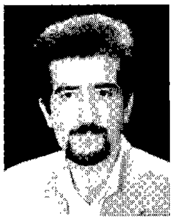
(9) W.J. Krzanowski and F.H.C. Marriott : Multivariate Analysis Part 2: Classification, Covariance Structures and Repeated Measurements, Edward Arnold, London (1994)

(10) S. Haykin : Neural Networsk, Prentice Hall, New Jersey (1999)

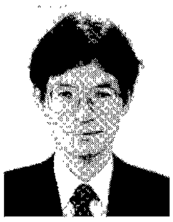
(11) N.Kambhatla : "Local Models and Gaussian Mixture Models for Statistical Data Processing", Ph.D. Thesis, Oregon Graduate Institute of Science & Technology (1996)

(12) R. Kohavi : "A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection", *Proc. of the 14th Int. Joint Conf. on Artificial Intelligence IJCAI*, pp. 1137-1145, Montreal, Canada (1995)

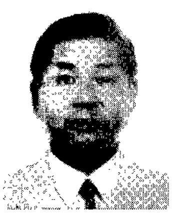
(13) K. Fukunaga : Introduction to Statistical Pattern Recognition, Academic Press, New York (1972)



Ali Ahmadi (Member) received B.Sc. in Electrical Engineering from AmirKabir University, Tehran, Iran and M.Sc. in Image Processing from Osaka Prefecture University, Japan in 1991 and 2001 respectively. He was working in field of computer programming and system designing from 1992-1998. Now he is studying in the field of Image Processing and Neural Networks as PHD candidate on Osaka Prefecture University. His research interests are Image Processing, Text Processing, Pattern Recognition and Neural Networks.



Sigeru Omatu (Member) received the B.E. degree in Electrical Engineering from the Ehime University in 1969 and the M.E. and Ph.D. degrees in Electronics Engineering from Osaka Prefecture University in 1971 and 1974, respectively. He was with the University of Tokushima as a research associate since 1974, a lecturer since 1975, an associate professor since 1980, and a professor since 1988. Then he joined the Osaka Prefecture University in 1995, where he is currently a professor in the Graduate School of Engineering.



Toshihisa Kosaka (Member) received the B.E degree in Electronics Engineering from Osaka Institute of Technology in 1970 and started working in Glory Ltd. at the same year. In 1999 he has received his Ph.D from Osaka Prefecture University. His field of research has been neuro-pattern classification for currencies. He is also member of IEE of Japan and the society of industrial and control engineers.