Malaysian Journal of Computer Science, Vol. 14 No. 1, June 2001, pp. 83-94

MULTICLASS TEST FEATURE CLASSIFIER FOR TEXTURE CLASSIFICATION

Itqon
Information Technology Department
Information Science Research Center
Tokyo 108-0075 Japan
email: itqon@isrc.co.jp

Shun'ichi Kaneko and Satoru Igarashi
Faculty of Engineering
Hokkaido University
Sapporo 060-8628 Japan
email: kanekos@coin.eng.hokudai.ac.jp

Vakthang Lashkia
Faculty of Engineering
Okayama University of Science
Okayama 700-0005 Japan
email: lashkia@ice.ous.ac.jp

ABSTRACT

A new multi-class pattern classifier called 'Test Feature Classifier' is presented. It is based on training a recogniser by training samples of binary patterns and voting primitive scores depending on many trained templates called 'test feature', which serves as local evaluation of the features. The method is non-metric and does not misclassify any patterns once learned previously. The two-class version of test feature classifier was of high performance for searching textual region in complex images. In this paper, we extend it to handle multi-class problems and apply it for solving ill-class problems in texture classification. We show the performance of the classifier on more than 1000 real images and compare it with a linear distance-based classifier and a non-linear distance-based classifier. The experimental results of both simulations and real applications show that the proposed classifier has better performance than conventional ones.

Keywords: ***Pattern recognition, Test feature classifier, Ill-class problem, Texture classification, Rank feature***

1.0 INTRODUCTION

The essence of classifier design can be said as the generation of discriminant functions in a feature space by utilising a priori knowledge of class distributions or given training samples [1, 2]. Many classifiers such as distance-based classifiers or Bayesian classifiers assume their classes as some probabilistic distributions. They do not have excellent capability to create discriminant functions more complicated than hyper-planes or at most quadratic surfaces for multi-dimensional feature space, so misclassification often occurs when the classes do not separate each other perfectly. Furthermore, in the case of no a priori knowledge on the classes or very small training samples, we cannot have reasonable model of their distributions. Here let us call such a problem as an ill-class problem.

In order to solve such ill-class problems, many classifiers have been proposed. Piecewise linear classifiers [1] permit to use some training samples for defining each class and then the discriminant surface consists of connected hyper-planes. We have to carefully select the representative samples to expect the good performance. Decision tree classifiers [3] can generate a complex discriminant by hierarchical threshold. However, the characteristics of features should be well understood to obtain the optimal threshold values. Neural networks [4] also need plural training patterns for feeding to back propagation-learning process to define the weights. Support vector classifiers [5, 6] utilise an extended feature space from the original space to construct a linear classifier in the extended space which may separate different classes, but extension of feature space derives another computation problem.

The ill-class problem also occurs when we hope a single distribution as a model of each class, which implicitly includes more than one cluster. As an alternative approach to the problem, we can classify each subclass as an independent class; otherwise we have to design some effective features, which generate no subclasses. This is one of the reasons why there have been many studies on feature extraction which are invariant to scaling, rotation, contrast change [7, 8, 9], etc. In short, the solutions to decrease misclassification also have been tried by feature-based approaches. However, approaches depending excessively on features are not suitable for an automatic system and it is not a simple task to design appropriate features.

We have proposed a novel pattern classifier called 'Test Feature Classifiers' (TFC) which can solve the ill-class problems without redesign of some special features, because TFC has a capability to create the complicated discriminant functions [10, 11]. TFC consists of two main procedures: *test feature extraction* and *voting selection*. The first procedure can be considered as a learning process where a test feature (or simply, a test) is a local combination of any features that can sufficiently distinguish the classes. The concept of test is an evaluation of several combinations of features (local recognition) to decide the result of entire features (global recognition). It was first introduced in [12] for the purpose of digital logic circuit analysis. The tests in TFC are generated directly from some training samples that are represented by bit patterns. In the second procedure, an unknown object will then be classified by voting primitive scores calculated from the tests. This method is non-metric and does not misclassify any patterns once learned previously. We have shown that TFC is of high performance for two-class problems by using it for recognition of character and non-character regions [13, 14, 15]. In this paper, we extend it to handle multi-class problems and apply it to both simulated and real images data. We also propose a new feature for texture analysis called rank feature [11].

This paper is organised as follows: Section 2.0 describes a mathematical formalisation of the proposed classifier and an artificial example is also given. The simulations to inspect the performance of TFC are presented in Section 3.0. In Section 4.0, after the description of rank feature, we describe the experiments of texture classification on more than 1000 real images. The performance of TFC is compared with the linear and non-linear distance-based classifiers. The experimental results of both simulations and the real texture classification show that TFC has better performance than the conventional ones. Section 5.0 concludes the paper with several concluding remarks.

2.0 MULTICLASS TEST FEATURE CLASSIFIER

The original idea of TFC is a discrete procedure of recognition, which is proposed in [16]. We then develop and formulate it as a general learning method (classifier) for pattern recognition. The voting functions are *m*-degree polynomials and can be used for partitioning the *n*-dimensional feature space ($m < n$), whose features are assumed to be binary valued. Statistical, structural or metrical characteristics of patterns are not required. We theoretically formulate TFC and then give an example of application to simple artificial data.

2.1 Formalisation

Suppose that $F = \{\bar{f} = (f_1, f_2, \cdots, f_n)$ is an n-dimensional feature space, and each pattern is represented as a binary-valued feature vector in this space $f_i \in \{0,1\}$. Let us also suppose that there is a collection of disjoint *h*-classes $\boldsymbol{C} = \{C_1, C_2, \cdots, C_h\} (h \geq 2)$ to be classified. Each class has a set of training samples as follows:

$$C_i = \{ {}_i\bar{x}_1, {}_i\bar{x}_2, \cdots, {}_i\bar{x}_{m_i} \} \tag{1}$$

where

$${}_i\bar{x}_j = ({}_ix_1^j, {}_ix_2^j, \cdots, {}_ix_n^j), \quad j = 1,2,\cdots, m_i \tag{2}$$

and

$${}_ix_k^j \in \{0,1\}, \; k = 1,2,\cdots, n \tag{3}$$

A collection of features $t = \{k_1, k_2, \cdots, k_g\}$ $(1 \leq g \leq n)$ is called a *test feature* (or simply, a test) if it satisfies the following conditions: for any different pair of classes u and v ($\neq u$), and for any p_u ($1 \leq p_u \leq m_u$) and p_v ($1 \leq p_v \leq m_v$), there always exists some feature bits such that

$${}_ux_{k_s}^{p_u} \neq {}_vx_{k_s}^{p_v} \tag{4}$$

The conditions certify that if we have a test we can always discriminate each class only by checking difference among the tests. A test $\boldsymbol{t} = \{k_1, k_2, \cdots, k_g\}$, is represented by an *n*-tuple vector $\bar{\boldsymbol{t}} = \{\boldsymbol{t}_1, \boldsymbol{t}_2, \cdots, \boldsymbol{t}_n\}$, where

$$t = \begin{cases} 1 & if \quad k \in t \\ 0 & otherwise \end{cases} \tag{5}$$

Then let us define the maximal set of tests derived from $\boldsymbol{C}$ as $T(\boldsymbol{C}) = \{\bar{\boldsymbol{t}}_1, \bar{\boldsymbol{t}}_2, \cdots, \bar{\boldsymbol{t}}_l\} (1 \leq l \leq 2^n - 1)$. We know there are all 2^n -1 combinations of possible features. From the extracted set of tests $T(\boldsymbol{C})$, we select *prime-test features* which are the tests whereby $\boldsymbol{t} - \{k_s\}$ do not satisfy the test condition for any s. In other words, an irreducible set of features which forms a test is called a prime-test, so that the set of tests $T(\boldsymbol{C})$ and the set of prime-tests $P(\boldsymbol{C})$ have a relation as follow:

$$P(\boldsymbol{C}) \in T(\boldsymbol{C}) \tag{6}$$

We call the above procedure as a *test feature extraction* that corresponds to a learning process.

In order to classify an unknown pattern $\bar{t}$, a *voting selection* is performed by calculating a score for each class as the following voting function:

$$V_i(\bar{t}) = \frac{1}{m_i} \sum_{t \in P} \sum_{j=1}^{m_i} \prod_{k=1}^{n} (1 - \boldsymbol{t}_k \left| t_k - {}_i x_k^j \right|) \tag{7}$$

Finally, the number i that corresponds to $\max\left(V_i(\bar{t})\right)$ shows the recognised class.

Since for all training samples ${}_i\bar{x}_k \in C_i$,

$$V_j({}_i\bar{x}_k) \begin{cases} \geq 1 & if \quad i = j \\ = 0 & otherwise \end{cases} \tag{8}$$

we can find an important property that TFC classifies all training samples correctly.

2.2 Numerical Example

It is easy to understand TFC by a simple numerical example. As shown in Table 1, consider that there are three classes on a four-dimensional feature space to be classified, where each class has two training samples represented in four bits (four-dimensional vectors). From this table, we can find that the first training sample of the second class, ${}_2\bar{x}_1$, is 0100, where ${}_2x_1^1$ is 0, ${}_2x_2^1$ is 1, and so on.

Table 1: Training samples

Class	Samples
C_1	0 0 1 0
	0 0 1 1
C_2	0 1 0 0
	1 0 0 1
C_3	1 1 0 1
	1 1 1 0

Since we have the four bits, there are 15 possible candidates of test. Tests and prime-tests can be detected as shown in Table 2. In order to avoid misunderstanding between vectors of training samples and vectors of tests, let us denote ○ and ● for vectors of tests where ○ is a referred bit and ● is an unreferred bit. For example, ●○●○ cannot be a test, because the training sample ${}_1x_2$ (0011) partly equals to the training sample ${}_2x_2$ (1001) due to the template ●○●○. On the other hand, since a combination of the second, third and fourth bits of all training samples does not have the same patterns, ●○○○ holds as a test and also as a prime-test. For another example, ○○●○ is not a prime-test although it is a test, because it is still satisfying as a test when the fourth component is taken away, i.e. then it will be ○○●● which is still a test. For training samples of Table 1, two prime-tests ●○○○ and ○○●● are obtained.

Next, consider that there is an unknown object represented as 1100. Table 3 shows the voting results of TFC for the object according to the extracted prime-tests. A prime-test can be considered as a template, which is put on all of training samples and the object. For example, by the template ●○○○, the first training sample of the second class (0100) will be -100, and the object will also be -100 ('-' is don't care bit). Thus, 1100 partly corresponds to 0100 due to the template; so one score is voted to the class of 0100. This calculation is executed for all training samples and all prime-tests. Finally, the total amount of scores are then normalised by the number of training samples of each class, and the class which gets the maximum score is to be the solution. Table 3 shows that TFC classifies the unknown 1100 as a member of C_3.

Table 2: Test and prime-test extraction

Test Candidate	Notes
●●●○	Not test
●●○●	Not test
●●○○	Not test
●○●●	Not test
●○●○	Not test
●○○●	Not test
●○○○	Test and prime-test
○●●●	Not test
○●●○	Not test
○●○●	Not test
○●○○	Not test
○○●●	Test and prime-tes t
○○●○	Test but not prime-test
○○○●	Test but not prime-test
○○○○	Test but not prime-test

Table 3: Voting result for unknown pattern '1100'

		Prime-Tests		Total
Class	Samples	●○○○	○○●●	Scores
	0 0 1 0	0	0	$\frac{0}{2} = 0.0$
C_1	0 0 1 1	0	0	
	0 1 0 0	1	0	$\frac{1}{2} = 0.5$
C_2	1 0 0 1	0	0	
	1 1 0 1	0	1	$\frac{2}{2} = 1.0$
C_3	1 1 1 0	0	1	

3.0 SIMULATION

For simulation, we have synthesised the artificial data on a region of 16 $\times$ 16 square. From the 256 possible patterns of the eight-dimensional feature space (n = 8), we have chosen 10 patterns per class randomly for training samples, and all patterns have been recognis ed. The results of recognition are plotted in the figures. The performance of TFC is compared with both linear and non-linear minimum distance-based classifiers. We have adopted the Euclidean distance for Linear Classifier (LC) and the Mahalanobis distance for Non Linear Classifier (NLC). For both LC and NLC, we used the coordinates of patterns as features. For TFC, we used a binary representation of the coordinates of patterns. So each training sample is represented in eight bits.

As shown in Fig. 1 and Fig. 2, since the discriminant boundaries created by either LC or NLC are not more complicated than a straight line or a quadratic line, misclassification of training samples has occurred. In Fig. 2, the three classes in the peripheral part consist of several clusters while the central class has only one cluster. The recognition results show that the dis tance-based classifiers recognise one class as one cluster. However, TFC allows one class to include several different clusters. If we consider that a cluster represents a subclass, it means that TFC can unify some different subclasses as a single class. This suggests that TFC may be useful for ill-class problems especially in the case of existence of subclasses.

4.0 TEXTURE CLASSIFICATION

Texture classification is one of the important research subjects in many fields. Recently, most of the studies on texture classification have been performed by the approaches of feature design [7, 8, 9, 17]. In this paper, we present the use of TFC for handling ill-class problems in the texture classification, especially the ill-class problem when one class is constructed from several independent subclasses.

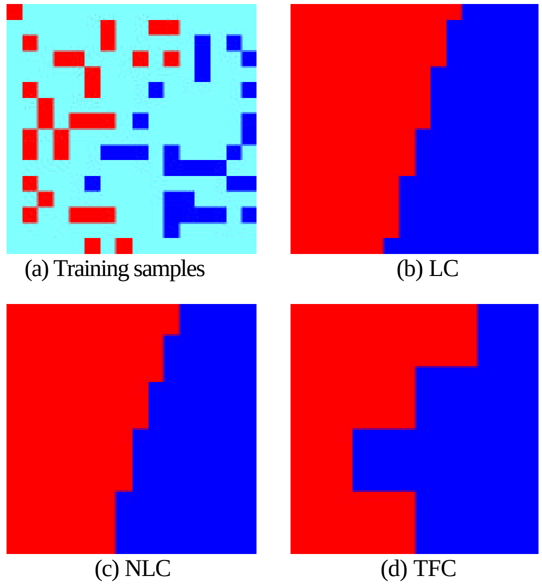


Fig. 1: Simulation for two-class problems

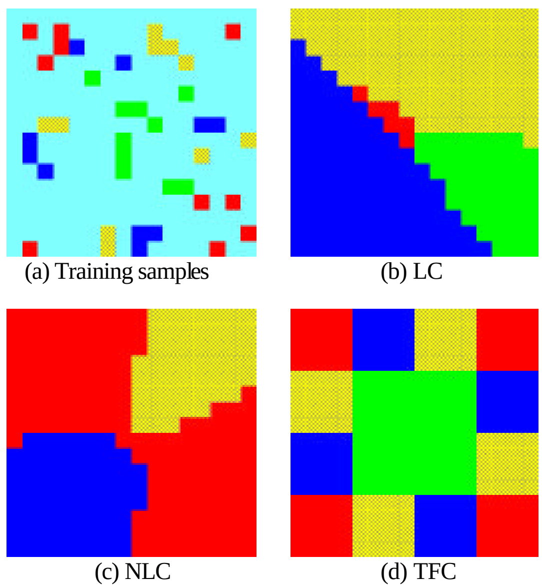


Fig. 2: Simulation for four-class problems

The subclass classification problems appear very often in many cases. For example in image analysis of remote sensing data [18], there are many kinds of water area such as sea, river, lake, pond, etc. Since their characteristics are different from each other, each of them might be classified as a different water area. However, in many cases, there maybe the requirement whereby they have to be recognised as the same area. In this case, many typical classifiers need to adopt the feature, which can unify them into a possible single representation. In other words, all of the water areas should be a single cluster when they are mapped on a feature space. A class in TFC should not be assumed as a single cluster, because TFC can recognise many disjoint clusters as a single class as we certified in the simulations.

Therefore, in this research, we adopt the co-occurrence matrix [19] which is one of the most popular features for texture analysis. Since they are real-valued features, we should encode them into binary codes, where some quantised parameters are necessary. In order to clarify this problem, the co-occurrence matrix are then transformed into what we call as *rank feature* which is a kind of the ordinal measure [20]. The robustness of this kind of measure has been confirmed for image correspondence [21].

4.1 Rank Feature

Suppose there is an image on a window where its intensity is expressed in N levels. The co-occurrence matrix $O(s)$ is an $N \times N$ matrix:

$$O = \begin{pmatrix} o_{11} & o_{12} & \cdots & o_{1N} \\ o_{21} & o_{22} & \cdots & o_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ o_{N1} & o_{N2} & \cdots & o_{NN} \end{pmatrix} \quad (9)$$

The o_{ij} represents the frequency of appearing of the intensity i and j simultaneously, where the pixel of j is located at a constant displacement s (angle and distance) from the pixel of i. From these matrix elements, some new features can be drawn such as energy, entropy, correlation, moment, etc [19]. We do not discuss them in detail since they are out of our focus.

Since the matrix O is symmetric ($o_{ij} = o_{ji}$), the matrix elements that should be calculated are only:

$$O' = \begin{pmatrix} o'_{11} & o'_{12} & \cdots & o'_{1N} \\ - & o'_{22} & \cdots & o'_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ - & - & \cdots & o'_{NN} \end{pmatrix} \quad (10)$$

where

$$o'_{ij} = \begin{cases} \frac{o_{ij}}{2} & if \quad i = j \\ o_{ij} & otherwise \end{cases} \quad (11)$$

The features of TFC are represented in binary-valued. However, if we adopt the coded version of o_{ij} or o'_{ij} as the feature furthermore, the binary representation has a long bit pattern unnecessarily. To avoid this drawback, we convert O' to the following $N \times N$ rank matrix *R:*

$$R = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1N} \\ - & r_{22} & \cdots & r_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ - & - & \cdots & r_{NN} \end{pmatrix} \quad (12)$$

where r_{ij} is the rank of element o'_{ij} among the matrix O'. The rank feature r_{ij} is always an integer with the range of $1 \le r_{ij} \le N(N+1)/2$, so all of them can be represented efficiently in the same length of bits, i.e. $\log_2(N(N+1)/2)$. From all the rank matrix elements, we arbitrary choose some fixed components, and use each bit of the encoded ones as a feature for TFC.

In a typical classification system, feature extraction is usually of key significance to the overall system performance. However, the performance of the system is dependent not only on the quality of the features, but also on the classifiability of the classifier. Thus, we use this simple rank feature to show that even with such a primitive feature TFC can achieve rather high and stable performance.

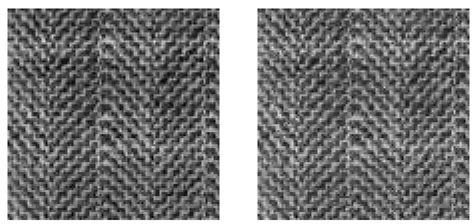
4.2 Experiments

The purpose of this experiment is to show that TFC has a capability to classify some independent subclasses as a single class. In real texture classification, the subclasses may appear when a class is generated from the similar texture images taken in different conditions of contrast, rotation or scaling. They will not be subclasses if we use some features, which are invariant with respect to them. The co-occurrence matrix and also rank features are contrast-variant features.

We deal with two cases of ill-class problems that come from the difference of brightness and the variety of different textures, respectively. In addition, as in the simulations, here we also compare the performance of TFC with LC and NLC.

4.2.1 Feature Extraction

In the experiments with real images, we have reduced the length of feature as follows. The intensity of the original images having 256 levels of gray-scale has been reduced into 8 levels by quantising with the equal interval of 32. This reduction only "roughens" the intensity but the characteristics of the textures are still preserved. Fig. 3 shows an example of the intensity reduction of a texture image. Thus, only 8×8 co-occurrence matrix can be calculated (in the experiments, we only adopt the co-occurrence matrix of constant displacement with 0 degree angle and 1 pixel distance). Each rank feature might be represented in five bits of binary code. From all of these features, we chose a combination of four features for each classification, getting a feature space of 20 dimensions. We use their binary representation for TFC while the unencoded values for LC and NLC. The original images are segmented into small sub-images of 64×64 pixels with no overlap as shown in Fig. 4. Each of these sub-images is considered as a texture pattern and used as one object for classification. We calculate the co-occurrence matrix from these sub-images.



(a) Original (256 levels) (b) Reduction (8 levels)

Fig. 3: Intensity-reduction of texture image

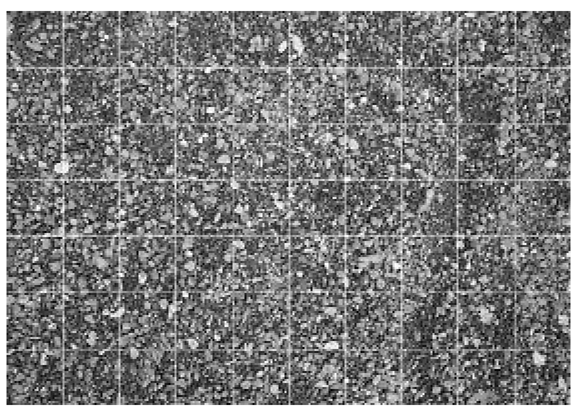
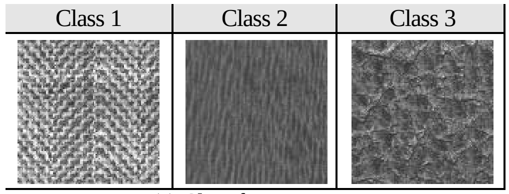


Fig. 4: Segmentation of original image

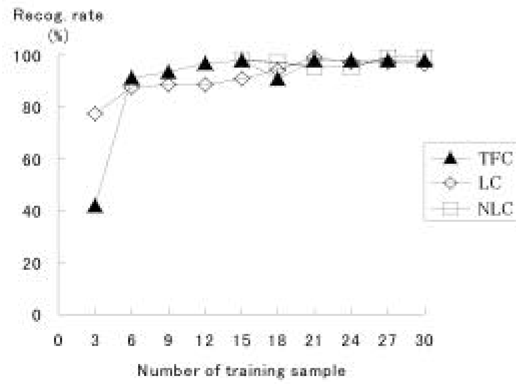
4.2.2 Ill-class Problem 1

We use three images of the well-known Brodatz texture down-loaded via a Web site [22]. From each image, two images with different brightness are created manually by performing multiplication of the original images with two different constants. Since the size of the original images is 512×512 pixels, we have obtained 384 sub-images for experiment. From each kind of images, 10 sub-images are selected randomly for training samples, and the remaining ones are used for recognition.

We have made two kinds of classification sets of training samples as shown in Fig. 5(a) and Fig. 6(a). Each class of the first classification set includes only the same texture images. The second classification set has the same textures except for brightness, so that an ill-class may appear since the rank feature is brightness variant. The maximum number of the extracted prime-tests for classification set 1 is 325 (24 training samples), the minimum number is 20 (3 training samples) and the average number is 181.9. For the classification set 2, the maximum is 542 (18 training samples), the minimum number is 83 (6 training samples) and the average number is 405.5.

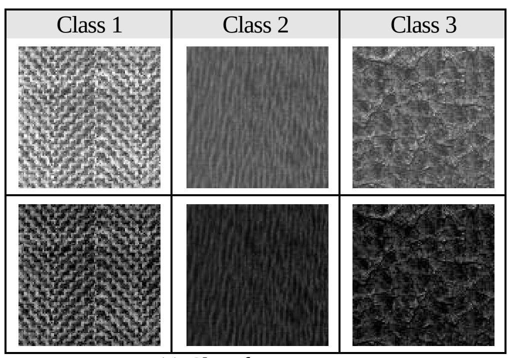


(a) Classification set 1

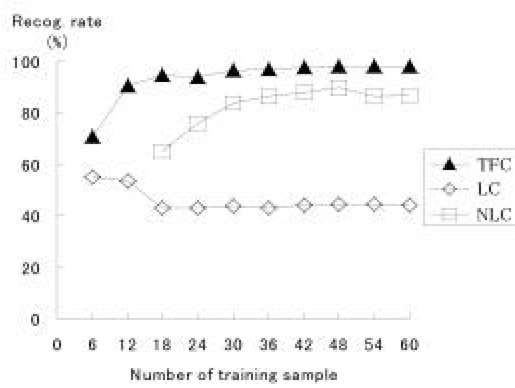


(b) Recognition results

Fig. 5: Recognition of classification set 1 for TFC, LC and NLC



(a) Classification set 2



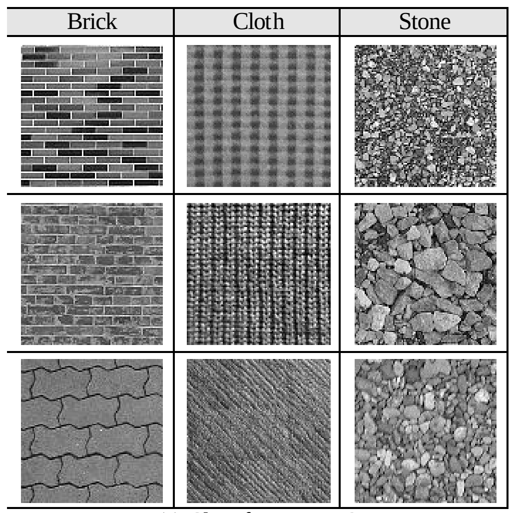
(b) Recognition results

Fig. 6: Recognition of classification set 2 for TFC, LC and NLC

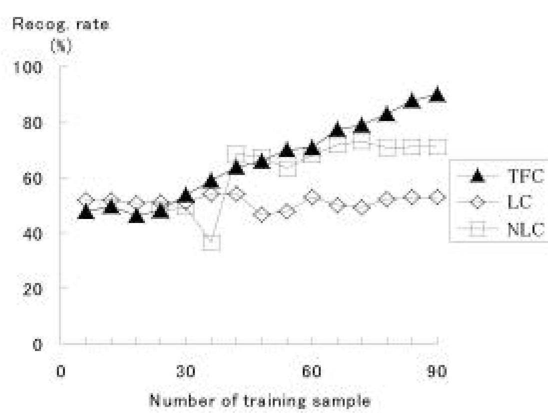
For classification set of Fig. 5(a), it can be considered that there might be no ill-class problem, therefore, all the classifiers could achieve the high recognition rates as we can see in Fig. 5(b). NLC cannot recognis e if the number of training samples is less than the number of features. On the other hand, TFC is effective although the number of training samples is not enough. As shown in Fig. 6(b), it is clear that LC can not handle this ill-class problem while NLC is not so bad, but their recognition rates are less than the one of TFC.

4.2.3 Ill-class Problem 2

In this second experiment, we have three classes of natural textures: brick, stone and cloth. The texture images are taken by a conventional digital camera. Each class includes subclasses corresponding to three kind of different textures as shown in Fig. 7(a). As in the experiment of previous section, we also segment the original images (640×480 pixels) into small texture images of 64×64 pixels, so that about 630 texture images are obtained from nine original images. From each kind of texture, we have selected 10 images for training samples (90 training samples in all) and the remaining ones are used for recognition. In order to show the effect of subclass in recognition result, we have divided the 90 training samples of Fig. 7(a) into three parts. The first part (from the first to the thirtieth) was the training samples from images of the first line of classification set 3. The second part (from the thirty first to the sixtieth) was from the second line and the third part (from the sixty first to the ninetieth) was from the third line. The maximum number of the extracted prime-tests for classification set 3 is 1542 (36 training samples), the minimum number is 53 (6 training samples) and the average number is 717.7.



(a) Classification set 3



(b) Recognition results

Fig. 7: Recognition of classification set 3 for TFC, LC and NLC

Fig. 7(b) shows the recognition results of classification set 3. For LC, the recognition rate worsens as the number of training samples increases. Although the recognition rate of NLC increases while the second part of training samples is learned, there is almost no change for the third part. In contrast with these results, the recognition rate of TFC increases monotonously as the number of training samples increases. It means that TFC has a capability to recognise some different subclasses as a single class. The recognition rate of TFC with 90 training samples is about 90%, however it may still increase if more training samples are given.

The recognition rates of TFC, LC and NLC for all the once-learned training samples of the classification set 1, 2 and 3 are shown in Fig. 8. It is clear that no misclassification of training samples occurs, only in the case of TFC.

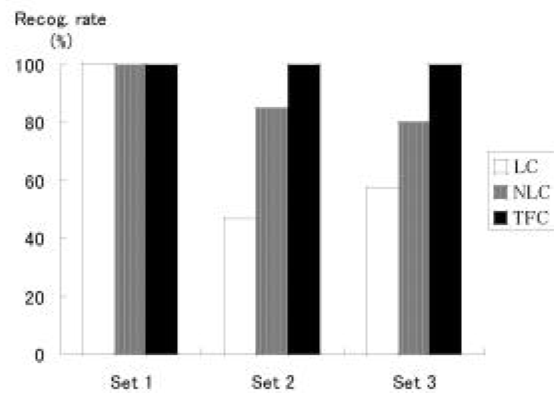


Fig. 8: Recognition of training samples

5.0 CONCLUSION

We have presented a novel multi-class pattern classifier called TFC. This classifier is based on constructing prime-test features, which are local combination of the features that sufficiently discriminate the classes of given training samples, and voting primitive scores depending on the prime-tests. Features utilised in TFC have binary-coded representations, by which the extraction of prime-tests and the computation of scores for voting can be performed easily and fast through bit operations. Another important characteristic is that it has no misclassification on any patterns learned previously.

The ability of TFC to create the complex boundary of discrimination was confirmed by the simulation. This suggests that it can effectively solve the ill-class problem that occurs when a class is constructed by some different subclasses. TFC has been applied to solve the ill-class problems in the real texture classification. We have inspected the performance of TFC on more than 1000 real texture images of the two cases of ill-class problems. From the comparative experiments with the linear and non-linear distance-based classifiers, TFC shows excellent performance in both simulation and real applications. We have also proposed a new kind of feature for texture classification called rank feature, that has good performance even for real texture images.

Since the voting selection is performed for all training samples and all prime-tests, a suitable number of prime-tests is very expected. Setting the importance of each prime-test may reduce the number of prime-tests. We left this problem as the next work.

ACKNOWLEDGMENT

We wish to thank Prof. S. Aleshin from Moscow State University for his helpful discussion and comments for this research.

REFERENCES

[1] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. John Wiley & Sons, New York, 1973.

[2] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, New York, 1972.

[3] S. Shlien, "Multiple Binary Decision Tree Classifiers". *Pattern Recognition*, Vol. 23, 1990, pp. 757-763.

[4] R. Lippmann, "Pattern Classification Using Neural Networks". *IEEE Communication Magazine*, Nov. 1989, pp. 47-64.

[5] C. Cortes and V. Vapnik, "Support-Vectors Networks". *Machine Learning*, Vol. 20, 1995, pp. 273-297.

[6] M. Pontil and A. Verri, "Support Vector Machines for 3D Object Recognition". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-20, 1998, pp. 637-646.

[7] R. L. Kashyap and A. Khotanzad, "Model-Based Method for Rotation Invariant Texture Classification". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, 1986, pp. 472-481.

[8] Y. Yoshida and Y. Wu, "Classification of Rotated and Scaled Textures Images Using Spectral Moments", in *Proc. of IAPR Workshop on Machine Vision Applications*, 1996, pp. 275-278.

[9] T. N. Tan, "Rotation Invariant Texture Features and Their Use in Automatic Script Identification". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-20, 1998, pp. 751-756.

[10] Itqon, S. Kaneko, V. Lashkia and S. Igarashi, "Multi-Class Test Feature Voting for Texture Classification", in *Proc. of the 4th International Conference on Quality Control by Artificial Vision*, 1998, pp. 55-62.

[11] Itqon, V. Lashkia, S. Kaneko and S. Igarashi, "Test Feature Voting Classifier for Ill-class Problems in Texture Classification", in *Proc. of the 5th Korea-Japan Joint Workshop on Computer Vision*, 1998, pp. 62-67.

[12] I. Chegis and S. Yablonsky, "Logical Methods for Controlling Electric Circuit Function", in *Proc. of V.A. Steklov Institute of Mathematics*, Vol. 51, 1958.

[13] Y. Hayakawa, S. Kaneko, V. Lashkia and S. Igarashi, "Image Recognition by Instance Voting Method Using Fourier Features", in *Proc. of the 3rd Symposium on Sensing via Image Information*, 1997, pp. 133-138 (in Japanese).

[14] V. Lashkia, S. Kaneko and S. Aleshin, "A Novel Approach to Locating Textual Region in Complex Images", in *Proc. of the IEEE Workshop on Robot and Human Communication*, 1998, pp. 537-542.

[15] V. Lashkia, S. Kaneko and S. Aleshin, "Textual Region Location in Complex Images Using Test Feature Classifiers". *Canadian Journal of Electrical and Computer Engineering*, Vol. 24, No. 2, 1999, pp. 65-71.

[16] S. Aleshin, *Metrization of Discrete Procedure of Recognition*. Moscow University Press, 1989 (in Russian).

[17] U. Michael, "Sum and Difference Histograms for Texture Classification". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, 1986, pp. 118-125.

[18] K. Dony, K. Fukue, H. Shimoda and T. Sakata, "A Spatial Land Cover Classification with the Aid of Neural Network for Multi Temporal High Resolution Satellite Images". *Journal of the Japan Society of Photogrammetry and Remote Sensing*, Vol. 34, 1995, pp. 14-24.

[19] R. M. Haralick, "Statistical and Structural Approach to Texture", in *Proc. IEEE*, Vol. 67, 1979, pp. 786-804.

[20] M. Kendall and J. D. Gibbons, *Rank Correlation Methods* (Fifth edition). Edward Arnold, London, 1990.

[21] D. N. Bhat and S. K. Nayar, "Ordinal Measures for Image Correspondence". *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1998, Vol. PAMI-20, pp. 415-423.

[22] http://www.stanford.edu/class/cs328b/images/brodatz.

BIOGRAPHY

Itqon received the B.S. degree in information engineering and the M.S. degree in bio-application and system engineering from Tokyo University of Agriculture and Techno logy, Japan, in 1996 and 1998, respectively, and then the Doctor of Science degree in information engineering from Hokkaido University in 2001. Currently, he is a researcher at Information Science Research Center Co. Ltd., Japan. He specialises in non-parametric pattern recognition.

Shun'ichi Kaneko received the B.S. degree in precision engineering and the M.S. degree in information engineering from Hokkaido University, Japan, in 1978 and 1980, respectively, and then the Ph.D. degree in systems engineering from the University of Tokyo, Japan, in 1990. He had been a research assistant of the Department of Computer Science since 1980 to 1991, an associate professor of the Department of Electronic Engineering since 1991 to 1995, and an associate professor of the Department of Bio-application and Systems Engineering since 1996 to 1996, in Tokyo University of Agriculture and Technology, Japan. Currently, he is an associate professor of the Department of Control and Information Engineering in Hokkaido University from 1996. He received the Best Paper Award in 1990, the Society Award in 1998, respectively, from Japan Society of Precision Engineering. His research interests include machine vision, image sensing and understanding, robust image registration. He is a member of JSPE, IEICE, IPSJ and IEEE.

Satoru Igarashi received the B.S., M.S. and Ph.D. degrees in precision engineering from Hokkaido University, Japan, in 1963, 1965, and 1971, respectively. Currently, he is a professor of the Graduate School of Engineering in Hokkaido University. He received the Best Presentation Award from Japanese Society of Die and Mould Technology in 1995, and the Society Award from Japan Society of Precision Engineering in 1998. His research interests include production system, mechanics, and machine vision. He is a member of JSPE, SGE and the Society of Life Support Technology.

Vakthang Lashkia received the M.S. and Doctor of Science degrees in computer science, from Moscow State University, in 1984 and 1988 respectively. Currently, he is an associate professor of Okayama University of Science. His previous working places are: Moscow State University, Tbilisy University, and Hitotsubashi University. His research interests include machine learning, pattern recognition and automata theory. He is a member of IEEE and ACM and IEICE.