MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan

6-1

A Hierarchical Classification Method for US Bank Notes

Tatsuhiko Kagehiro [1], Hiroto Nagayoshi [1], Hiroshi Sako [1]

Central Research Laboratory,
Hitachi, Ltd.,

Abstract

This paper describes a method for the classification of bank notes. The algorithm has three stages, and classifies bank notes with very low error rates and at high speeds. To achieve the very low error rates, the result of classification is checked in the final stage by using different features to those used in the first two. High-speed processing is mainly achieved by the hierarchical structure, which leads to low computational costs. In evaluation on 32,850 samples of US bank notes, with the same number used for training, the algorithm classified all samples precisely with no error sample.

1 Introduction

A lot of machines have to handle bank notes, including the automated teller machines (ATM) of banks, ticket-vending machines at railway stations, and vending machines for drinks, cigarettes, and so on. The bank-note validation systems of such machines consist of two functions: classification and verification. The type of bank note and the direction of input are distinguished by the classification function. The result of this process is used in verification. Since the points observed to verify a bank note differ with the type and direction of the note, the result of classification has to be precise. Furthermore, since the embedded processors of teller machines and so on are typically embedded RISC processors, the calculation cost of the algorithm has to be low.

Bank-note validation algorithms are typically not made public because of security concerns. There is thus little published work in this area. Among others, however, work on the classification of Italian lira notes has been published [1]. Learning vector quantization (LVQ) is adopted as a learning stage and the type of bank note is classified by a neural network. This method can classify the bank notes precisely, but the numbers of evaluation samples are small. Work on a method for the classification of Euro bank notes has also been published [2]. This method is based on a three-layered perceptron and radial basis function (RBF) networks, and has an automatic learning stage. Since the algorithm includes MAC (multiply and accumulate) operations, however, it is too computationally costly to be implemented on an embedded RISC processor. A method of validation that has been implemented on a real machine has also been described [3]. However, because this method uses information on the sizes of the notes, it is not suitable for classifying bank notes that have the same size (for example, the various denominations of US dollars).

[1] Address : 1-280, Higashi-koigakubo Kokubunji-shi, Tokyo 185-8601, Japan

E-mail : {kagehiro, hiroto-n, sakou}@crl.hitachi.co.jp

In this paper, we propose a method of classification which is purely based on information on the bank note, and carries a low computational cost.

We tested our bank note classification method on US bank notes. US dollars come in 12 denominations and are currently of 12 types: $1, $2, $5, $10, $20, $50, $100, new $5, new $10, new $20, new $50, and new $100. There are four possible input directions, as shown table 1. The classification module thus has to distinguish 48 categories, in the situation that each input direction is assumed as different category. US bills are particularly difficult to categorize, since they are all of the same size and have similar printed patterns.

2 Bank-Note Validation System

Figure 1 shows the validation system. The bank note is validated by using data from various sensors. Outputs are the type of bank note and the result of verification. The system consists of two modules: a bank-note classification module and a bank-note verification module. The classification module distinguishes the type and direction of input of the note, while the verification module differentiates authentic notes from counterfeits.

Since bank notes are usually used over several years, the classification module has to be robust to take deterioration over time into account. If points or areas for observation are selected from local parts of the bank note, the module will be overly sensitive to soiling of and wear on the bank note. Therefore, the points or areas have to be selected from all over the note. However, if we obtain a high-resolution image of the note and use all of its pixels in classification, the computational cost will be huge. To obtain both robustness and high speeds, a number of dis-

Table. 1 : Bank-note Directions

 Direction A	 Direction B
 Direction C	 Direction D

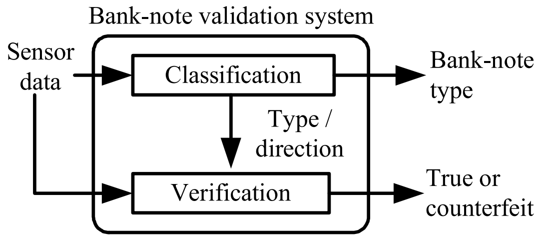

Figure. 1 : Bank-Note Validation System

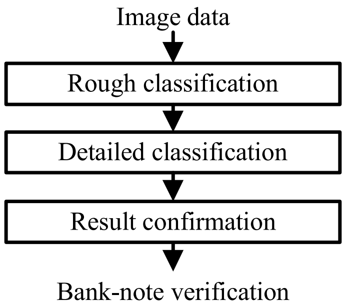

Figure. 2 : Classification Module

crete points are selected from the overall image, and the average of the pixel at each point and its adjacent pixels is taken as the observed value for each point. The average values are calculated by the hardware in advance of classification.

The classification module must be highly accurate, since the information on the type of bank note and its input direction determine the observation points for the verification module. If the result of classification is incorrect, these observation points will be invalid and the verification module will not be operating on the right data.

3 Classification Module

3.1 Overview of the Bank-Note Classification Module

The classification module must achieve both high-speed processing and high accuracy. Since the bank note is classified by measuring the distance between a template vectors and the feature vectors from the observation points for use in classification, high-speed processing is realized by decreasing the computational costs of this distance measurement. We achieve high speeds in the first two stages of our three-stage architecture. High accuracy is maintained by including a result-confirmation step as the third stage of classification. This result confirmation step uses different observation points from those used in the basic classification stages.

Figure 2 shows the flow of operations in the classification module. The process has three stages: rough classification, detailed classification, and result confirmation. In rough classification, several possible candidates are selected as correct answers; in detailed classification, a single candidate is selected. For low computational costs, low-dimensionality vectors are used in rough classification. Since the accuracy of this stage of classification is not very high, however, several candidates are output, only one of which will be correct. For high accuracy, high-dimensionality vectors are used in detailed classification. However, since measurements are only required for a few candidates, the cost of processing is low.

3.2 Selecting Observation Points

In developing the classification module, we have to select an optimal set of observation points. It is important to select points where the different types of bank note have different features. If we can select such points, the results of classification will be precise. Furthermore, the requirement for low computational costs means that we have to keep the number of points down.

The method for selection of the observation points is as follows. Firstly, average images of each type of bank note are created. Every pixel of each average image is then compared with the corresponding pixels of the other average images. The per-pixel difference values thus produced are accumulated on a difference maps . The difference map created by all combinations of each average image shows the difference points among the bank notes types. Next, observation points on the map are selected in descending order of pixel values, except that pixels in the vicinity of selected points are excluded. We can thus automatically select many optimal observation points on each type of bank note in each orientation.

3.3 Rough Classification

In rough classification, candidate bank-note types are obtained by using small numbers of observation points. Firstly, pixel values are gotten from some of the observation points, and the resulting set of values is regarded as the feature vector of the input bank note. A lot of feature vectors are collected from learning data in advance, and a template for each category is created by using a generalized learning vector quantization (GLVQ) algorithm [4]. When an unknown bank note is input to this system, the feature vector is extracted from the observation points. The Euclidian distance between the extracted vector and each of the templates is then calculated, and tokens representing the templates are then sorted in ascending order of distance. The top few templates are selected as candidates, and are used in the detailed classification step.

3.4 Detailed Classification

In detailed classification, a single correct template is selected from among the candidates by using large numbers of observation points. All of the templates have been created in advance by using the GLVQ algorithm. When an unknown bank note is input, the feature vector is extracted and the Euclidian distance between the feature vector and each of the candidate templates is calculated. The candidate template with the smallest distance is selected as the correct answer.

Since high-dimensionality vectors are used in this stage, the computational cost of each comparison is relatively large. However, since comparison is only with a few candidates, the total cost is quite small.

3.5 Result Confirmation

In this stage, the results of detailed classification are checked by using other features. Average values of a pixel and its nearest neighbors are used as feature vectors in both the rough and detailed classification stages; in the result-confirmation stage, values for individual pixels are used as micro-pattern information. This usage of different information makes the confirmation step effective.

Particular pixels from selected lines are examined in the result-confirmation stage. The lines for each type of bank note are independently determined in advance, so the selected lines generally vary with the result of classification. The selected lines are at positions that reflect the different features of the various types of bank note, and thus characterize the types of bank note. The set of pixel values is regarded as the feature vector. Feature vectors of the same type are collected from the learning data, and a template for each type of bank note is created in advance by using the GLVQ algorithm.

When the result of classification for a bank note of unknown type is input to the result-confirmation stage, the result of detailed classification determines the template to be used, and thus the lines for observation. The Euclidian distance between feature vectors for the note and the selected template is calculated. If this distance is below a certain threshold, the input bank note is accepted.

4 Experimental results

4.1 Evaluation on Samples of Real Bank Notes

We evaluated this bank-note classification module on US bank notes. A summary of the characteristics of the learning and evaluation data sets is given in table 2. The learning and evaluation data sets have the same size but consist of different notes.

The numbers of observation points and lines were determined in consideration of computational cost and are listed in table 3. The system produced no errors in validation of the evaluation data set.

And we measured that the classification module takes 16ms per one bank note on the embed RISC processor of 78 MIPS.

We need more samples to obtain an accurate error rate, but gathering sample bank notes is difficult. Instead, we estimated a global distribution for US bank notes from the results for our sample set, and calculated an error rate on the assumption that this estimate is valid.

In order to estimate a global distribution for US bank notes, huge numbers of bank note samples are needed with the object of the statistics. But we estimate the error rate on the assumption that we can prepare the huge amount of bank note data. This estimated error rate can be considered as the indicator which presents the classification performance. So we will have to judge the estimated error rate of this experiment as future works.

4.2 Estimating the Error Rate in Classification

In the classification step, we take the Euclidian distances between the feature vectors of the note being validated and the templates. We thus obtain an estimated distribution by taking the average and variance of the Euclidian distance for all samples with all templates, on the assumption that the distribution is Gaussian. This distribution is then used to estimate the error rate.

For example, we consider estimation of the error rate for the misclassification of bank notes in category A as category B. Firstly, all bank note samples of category A are input to the classification module. As a result, the Euclidian distance between the feature vector for each note and the category A template is calculated. We refer to these calculated distances as correct-case distances. Next, the distances between the feature vector for each note of category A and the template for category B are calculated. We refer to these distances as incorrect-case distances. We plot all distance data on a map, as shown in Figure 4, where the vertical axis is the correct-case distance and the horizontal axis is the incorrect-case distance. Since all of the correct samples fall into that portion of the map where the correct-case distance is smaller than the incorrect-case distance, we can regard the area below Y = X as the correct-case area and the area above this line as the incorrect-case area. We can then take the estimated bank-note distribution and consider the part which falls in the incorrect-case area to be the error rate (see Figure 4). That is, the area of this part is the estimate of the error

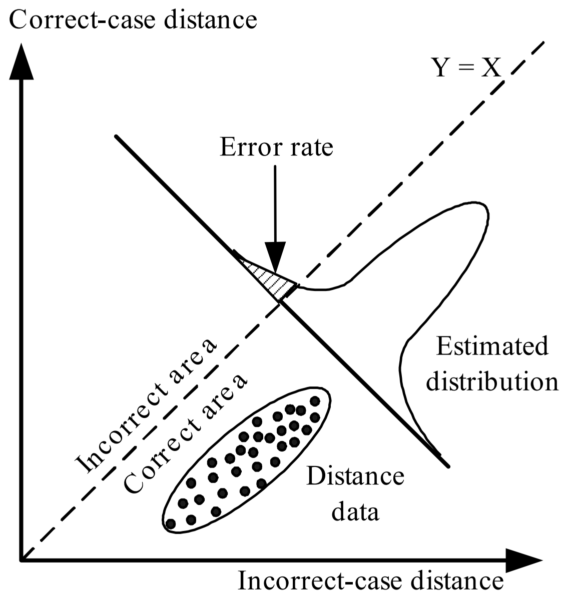

Figure. 3 : Error in Classification

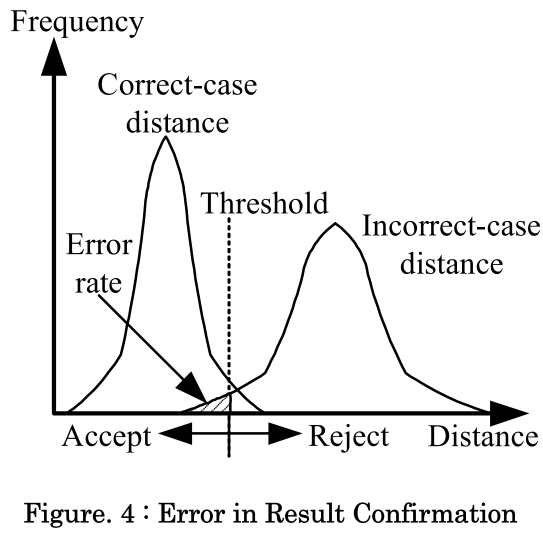

Figure. 4 : Error in Result Confirmation

Table 2: Experimental Data

Type of bank notes	US dollars
Learning data set	32,850
Evaluation data set	32,850
Number of types	12
Number of categories	48

Table 3: Classification Parameters

Rough classification	Number of points	20 (obverse: 10; reverse: 10)
	Number of candidates	3
Detailed classification	Number of points	120 (obverse: 60; reverse: 60)
Result confirmation	Number of lines	4 (obverse: 2 reverse: 2)

rate.

We used this method of estimation to calculate error rates for all combinations of bank-note type (for example, category A is $1, category B is $2, and so on).

4.3 Estimating the Error Rate in Result Confirmation

In result confirmation, the result of classification is checked by taking the Euclidian distance between the feature vector of the note being validated and that of the template. If the result of classification was correct, the bank note will match the template, so the sample should be accepted in this case. If the result of classification is incorrect, the bank note will not match the template, so the sample should be rejected. The respective distances between the feature vector for the note and the template are again referred to as the correct-case and incorrect-case distances.

We calculated the correct-case distances and incorrect-case distances in result confirmation for all bank-note samples. The distribution of each distance was then estimated from the average and variance of all samples. Figure 5 shows the distributions.

As stated above, a threshold for rejection is used in result confirmation. The incorrect-case area is thus only that part of the estimated incorrect-case distribution which falls below this threshold, as is shown in Figure 5. We regard this area as the error rate.

4.4 Total Error Rate Estimation

We now describe the estimation of the overall error rate for the classification module. If, for example, we are estimating the error rate for three categories (A, B, C), we start by estimating the total error rate for the situation where input bank notes of category A are misclassified as category B. This total rate is gotten by multiplying the rate of errors in classification by the rate of errors in result confirmation. Total error rates are then obtained for the other combinations (A is misclassified as C, B is misclassified as C). In this example, we take the highest of the three error rates as the error rate for the validation system.

Finally, we used these techniques to estimate the error rate for the validation system on our sample of US bank notes. We set the threshold of the result-confirmation stage so that 99% of all bank notes were accepted.

Among all combinations of bank-note types, the greatest error rate was 3.1E-9 for the misclassification of old $100 bank notes as $2 bank notes.

5 Conclusion

We have developed a bank-note classification module as part of a bank-note validation system. After training on 32,850 samples, the system produced no errors in evaluation on another 32,850 samples (samples were US notes). We estimated an error rate of less than 3.1E-9.

The algorithm used is hierarchical and the final stage of result confirmation is based on different information to that used in classification, so the system classifies bank notes quickly and precisely.

In future work, we will develop a bank-note verification algorithm which can be automatically customized adapted to new notes by being fed learning data. And we have to judge the estimated error rate of the classification by using more bank note samples.

References

[1]. T. Kosaka, et al., “Classification of the Italian Liras Using the LVQ Method”, International Joint Conference on Neural Networks 2000, 3, pp. 143–8 (2000).

[2]. M. Aoba, et al., “Euro Banknote Recognition System Using a Three-layered Perceptron and RBF Networks”, IPSJ Trans. Mathematical modeling and Its Applications Vol. 44, No. SIG7(TOM8), pp. 99-109, 2002.

[3]. A. Frosini, et al., “A Neural Network-Based Model for Paper Currency Recognition and Verification”, IEEE Trans. Neural Networks, Vol. 7, No. 6, pp. 1482–1490, 1996.

[4]. A. Sato and K. Yamada, “Generalized learning vector quantization”, Advances in Neural Information Processing Systems 8, MIT Press, Cambridge, MA, pp. 423–429, 1996.