Bit Plane Decomposition And The Scanning n-tuple Classifier

S. Hoque, K. Sirlantzis, M. C. Fairhurst
Department of Electronics, University of Kent, Canterbury, Kent CT2 7NT, United Kingdom.
E-mail: {msh4,ks30,mcf}@ukc.ac.uk

Abstract

This paper describes a multiple classifier configuration for high performance off-line handwritten character recognition applications. Along with a conventional Scanning n-tuple classifier (or sn-tuple) implementation, three other sn-tuple systems have been used which are trained using a binary feature set extracted from the contour chain-codes using a novel decomposition technique. The overall accuracy thus achievable by the proposed scheme is much higher than most other classification systems available and the added complexity (over conventional sn-tuple system) is minimal.

1. Introduction

Automatic recognition of printed and handwritten documents has been a challenging problem for many decades and many different methods have been reported. Although many of these show very high performance, none has been able to achieve the level of accuracy and speed shown by human readers, which may be considered the ultimate target. This implies that there is ample scope as well as necessity for improvement of the solution to this well-researched problem. This paper focuses on augmenting a simple high performance recognition scheme, the sn-tuple classifier, to further improve its recognition accuracy. A complete recognition system involves modules such as image capture, pre-processing, character segmentation, feature extraction, post-processing, etc. The experiments reported in this paper involve only the recognition phase of the pre-segmented characters.

The sn-tuple classification system has been shown to achieve very high recognition rates in character recognition applications at a reasonably high speed [10, 11, 12]. The augmentation proposed in this paper is to decompose the contour chain-codes already extracted by the original sn-tuple algorithm into three binary layers and use them to independently train (as well as test) three additional sn-tuple classifiers. A multiple classifier fusion strategy then combines the individual outcomes and generates an overall classification decision. The idea of decomposition is based on an approach proposed by Schwarz [14] as a means for compressing data. This decomposition technique has already been used successfully in manipulating gray scale face images for recognition [4].

The following sections of the paper first briefly describe the sn-tuple classifier and introduce the decomposition method used. The paper then proposes a multiple classifier scheme to combine the resulting classifiers, presenting results obtained over a series of cross-validating experiments. The paper ends with a brief conclusion.

2. The Scanning n-tuple Classifier (sn-tuple)

The Scanning n-tuple (or simply sn-tuple) classifier has been introduced by Lucas *et al.* as a statistical-syntactic method for high performance OCR applications [10, 11, 12]. This is a variant of the conventional n-tuple classifier except that instead of using the 2-D binary images directly, the operation is conducted on a 1-D gray scale representation of the bitmap image. Another difference between the n-tuple and the sn-tuple is that, whereas each n-tuple samples a set of fixed points in the input space, each sn-tuple defines a set of relative offsets between its input points. Each sn-tuple is then scanned over the entire input space.

The uni-dimensional model of the character image is obtained by tracing the contour edges of the image and representing the path by Freeman chain codes [3]. The sn-tuple algorithm can handle only one chain code string per pattern. Therefore, for images consisting of multiple contours, all strings are mapped to a single string by discarding the positional information (i.e., the start coordinates) and then concatenating the strings together. The length of a chain-code string is dependent on the character class as well as writing style, degree of slant, image quality etc. Since image classes with shorter chain-code description are often adversely affected in recognition, all these chains need to be expanded to a predefined fixed length before training and testing. Details of the sn-tuple classification algorithm (including pseudocode) can be found in [10, 11, 12].

Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR'02)
0-7695-1692-0/02 $17.00 © 2002 IEEE

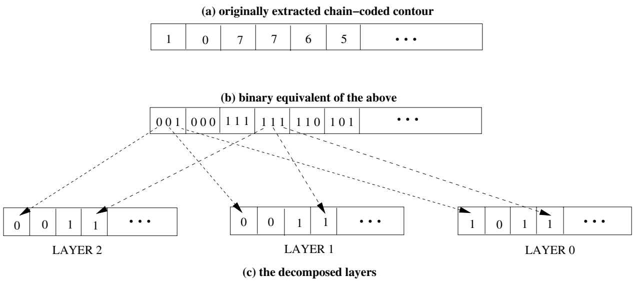

Figure 1. A Schematic of the Proposed Feature Decomposition.

3. Bit Plane Decomposition

The decomposition process is illustrated in Figure 1. The individual data elements (here, the Freeman direction codes) are represented in binary. Since there are 8 possible distinct direction codes, 3-bit binary numbers are sufficient to represent them. Now the chain-code string is decomposed into 3 separate strings (described as '*Layers*' in this paper) such that layer 'i' is composed only of the ith bits of the corresponding chain codes. Figure 1 illustrates how an arbitrary chain-code is decomposed into its equivalent 3 layers.

It is also possible to use other forms of binary notation (for example, Gray coding, Excess-3, etc.) to express the direction codes before decomposition and classifier performance is somewhat dependent on this choice (see [4] for details). For the experiments reported in this paper, the Gray coding was used for decomposition.

4. The Proposed System

The proposed system for handwriting recognition is a parallel combination of 4 sn-tuple based classifier implementations. The first sn-tuple implementation is the conventional one and is trained on the Freeman chain-code of the character images. Three further classifiers are trained on the decomposed layers of the already extracted Freeman chain-code string. Finally, a decision fusion stage than combines the outputs of the individual classifiers and generates a final class label for the test image.

5. Experimental Setup

Pre-segmented handwritten characters from an in-house database [16] have been used for the experiments. This particular dataset consists only of digits and uppercase letters with no distinctions made between '0'/'O' and '1'/'I' character pairs. There are 300 binary images for each character class, each of resolution 24×16 pixels. This dataset is randomly partitioned into two disjoint sets for training and testing. For the experiments reported here, 150 images per character class were used for the training. For the sake of simplicity as well as to keep computation to a minimum, all pattern normalization measures such as slant correction, broken image and stroke thickness mending, etc. were excluded. Experiments were conducted in two task domains, the first involving only the digits and the second involving all 34 alphanumeric characters. All the results reported in this paper are obtained through a series of cross-validation experiments.

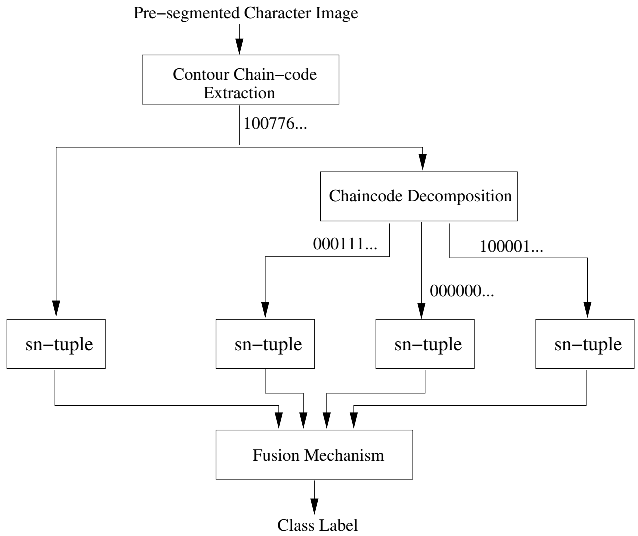

Figure 2. A Schematic of the Proposed System.

6. Results and Discussion

Before any fusion techniques were attempted, the performance of the individual sn-tuple classifier implementations was investigated. For ease of labelling, these classifiers are denoted SNT-M, SNT-L0, SNT-L1 and SNT-L2. SNT-M is the sn-tuple using the Freeman chain-code and the rest are using the layered chain-codes.

Table 1 shows the recognition error rates achieved by these implementations for the two task domains. It is readily evident that the performance of the original sn-tuple (SNT-M) is far superior compared to the others and it may be concluded that significant discriminatory information has been lost by the decomposition.

Although the sn-tuple classifiers using the decomposed layers (SNT-L0, SNT-L1, SNT-L2) showed very high error rates compared to SNT-M, it should be noted that memory space requirements by these implementations are very low (in the order of 2^n instead of 8^n of SNT-M, n is the tu-

Table 1. Performance of the different sn-tuple implementations in the proposed system

Classifier	Classification Error Rates (in %)	
	Numerals	Alphanumerics
SNT-M	4.59	12.42
SNT-L0	23.89	44.92
SNT-L1	21.28	40.68
SNT-L2	22.45	42.09

Table 2. Effect of fusion of the sn-tuple classifier operating on the decomposed layers only (SNT-L0, SNT-L1, SNT-L2)

Fusion rules applied	Classification Error Rates (in %)	
	Numerals	Alphanumerics
Mean	**11.41**	**23.69**
Median	13.07	28.30
Min	14.03	30.55
Max	19.65	40.96
Majority Vote	15.73	34.97

ple size). This leads to the next set of experiments where only the outputs of these three implementations are combined using various statistical multi-expert decision fusion rules. The five fusion schemes investigated are *mean, median, min, max,* and *simple majority vote*. Details of these fusion rules can be found in [9, 8]. Table 2 presents the combined error rates achievable using the decomposed layers by the sn-tuple classifier. The 'mean'-rule fusion (also known as the *sum*-rule) produced the lowest error.

Although it is seen that combining SNT-L0, SNT-L1, SNT-L2 can offer significantly better recognition accuracy than the individual configurations, the best of combined error rate is still more than twice that achieved by SNT-M. In the light of this, the next investigation is conducted using all four classifiers in the fusion. Since the 'mean'-rule fusion generated the optimum error rates in the previous fusion tests, only the 'mean'-rule fusion is used in the next experiment.

Table 3 shows the different classification error rates achieved by combining the SNT-M with one or more of the other sn-tuple implementations. The lowest error rates (shown in boldface) thus achieved are 1.41% and 5.44% respectively for the two task domains when SNT-M, SNT-L0 and SNT-L1 are combined. The reduction in error rates is significant (reduces to less than 1/3rd for 10 class and to less than 1/2 for 34 class tasks).

Figures 3 and 4 illustrate a comparison of mean error rates achieved by different classification schemes as tested in cross-validated experiments using the same database. In

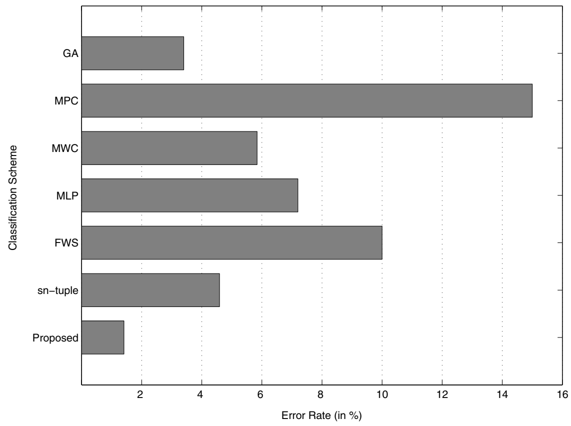

Figure 3. A Comparison of Achievable Error Rates (Numeral classification).

addition to the lowest error rates of the proposed architecture, error rates achieved by five other classifiers are shown. The conventional sn-tuple scheme has already been described in this paper. The FWS [2] is a frequency-weighted n-tuple classifier. The MLP is a standard multi-layer perceptron [7] trained using Back-propagation learning. The reported MLP had 40 hidden nodes and used zonal pixel density as the input feature. The MPC is a maximum-likelihood classifier [13] which explores possible cluster formation with respect to a distance measure. The implementation reported here uses Mahalanobis distance metric and up to 7th order geometric moments as features. The MWC [1, 5, 6] is an n-tuple based system where features are extracted from a sub-image isolated by a scanning window. The reported results are based on an MWC using a 21×13 pixel window and 12-tuples. It is readily evident that the proposed layout is capable of producing very low error classification decisions.

The schemes proposed here also performed favourably with respect to a trainable parallel multiple classifier system optimized using genetic algorithm (GA) techniques [15]. The minimum error rate achieved by this optimized system on the same database used here was 3.4% for the 10 class problem and 8.2% for the 34 class case.

7. Conclusion

This paper presents a novel pre-segmented character classification system consisting of a number of sn-tuple classifiers using a simple decision fusion scheme. One of these is the conventional sn-tuple classifier and the rest use decomposed chain-code as their feature spaces.

The merits of using sn-tuple with decomposed chain-code string are manifold. Although extraction of the chain

Table 3. Mean error rates of the proposed system for different combination strategies

Fusion Strategies	Classification Error Rates (in %)	
	Numerals	Alphanumerics
SNT-M + SNT-L0	2.11	7.61
SNT-M + SNT-L1	1.55	6.32
SNT-M + SNT-L2	1.76	7.17
SNT-M + SNT-L0 + SNT-L1	**1.41**	**5.44**
SNT-M + SNT-L0 + SNT-L2	1.92	6.31
SNT-M + SNT-L1 + SNT-L2	1.52	6.42
SNT-M + SNT-L0 + SNT-L1 + SNT-L2	2.13	6.48

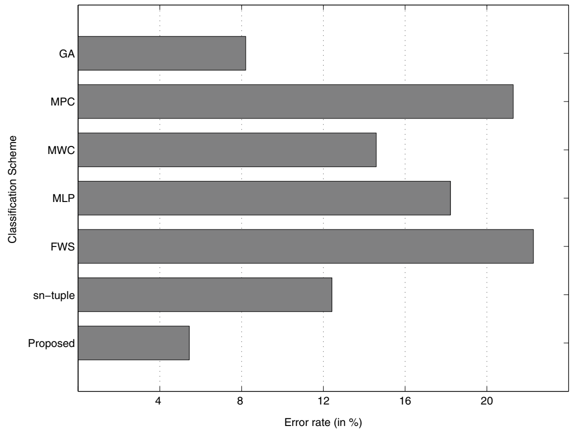

Figure 4. A Comparison of Achievable Error Rates (Alphanumeric classification).

code from off-line images is computationally expensive, the classification phase is very fast. Extraction of layers from the chain-code is through simple logic operations, and hence the additional computational load in feature extraction is minimal. Therefore, overall computational time will increase only by a nominal amount compared to the conventional sn-tuple implementation.

The other bottleneck of using an n-tuple based system is that such approaches demand the availability of a very large physical memory. The situation worsens when the system needs to work with gray-scale images (which is analogous to the chain-code string used in sn-tuple). For the conventional sn-tuple system, the memory demand is of the order of 8^n. For the sn-tuple system using the proposed decomposed chain-code string, the demand is in the order of 2^n. Therefore, by introducing the additional sn-tuple classifiers in the system, the overall memory demand will grow only by about 2-3%. Therefore, the increase in the computational complexity in the proposed layout is minimal although the improvement in error rates in very high.

Acknowledgments

The authors gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council.

References

[1] M. C. Fairhurst and M. S. Hoque. Moving window classifier: Approach to off-line image recognition. *Electronics Letters*, 36(7):628–630, March 2000.

[2] M. C. Fairhurst and T. J. Stonham. A classification system for alpha-numeric characters based on learning network techniques. *Digital Processes*, 2:321–329, 1976.

[3] H. Freeman. Computer processing of line-drawing images. *ACM Computing Surveys*, 6(1):57–98, March 1974.

[4] M. S. Hoque and M. C. Fairhurst. Face recognition using the moving window classifier. In *Proceedings of 11th British Machine Vision Conference (BMVC2000)*, volume 1, pages 312–321, Bristol, UK, September 2000.

[5] M. S. Hoque and M. C. Fairhurst. A moving window classifier for off-line character recognition. In *Proceedings of 7th International Workshop on Frontiers in Handwriting Recognition*, pages 595–600, Amsterdam, The Netherlands, September 2000.

[6] M. S. Hoque and M. C. Fairhurst. An improved learning scheme for the moving window classifier. In *Proceedings of 6th International Conference on Document Analysis and Recognition*, pages 607–611, Seattle, Washington, USA, September 2001.

[7] A. K. Jain, J. Mao, and K. M. Mohiuddin. Artificial neural networks: A tutorial. *Computer*, pages 31–44, March 1996.

[8] J. Kittler. Combining classifiers: A theoretical framework. *Pattern Analysis and Application*, 1:18–27, 1998.

[9] J. Kittler, M. Hatef, R. P. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, March 1998.

[10] S. Lucas and A. Amiri. Recognition of chain-coded handwritten character images with scanning n-tuple method. *Electronic Letters*, 31(24):2088–2089, November 1995.

[11] S. Lucas and A. Amiri. Statistical syntactic methods for high performance ocr. *IEE Proceedings Vision, Image and Signal Processing*, 143(1):23–30, February 1996.

[12] S. M. Lucas. Improving scanning n-tuple classifiers by pre-transforming training data. In *Proceedings of International Workshop on Frontiers in Handwriting Recognition-V*, pages 143–146, ESSEX, UK, 1996.
[13] R. J. Schalkoff. *Pattern Recognition: Statistical, Structural and Neural Approaches*. John Wiley and Sons, Inc, 1992.
[14] J. W. Schwarz and R. C. Barker. Bit-plane encoding: A technique for source encoding. *IEEE Transaction on Aerospace and Electronic Systems*, 2(4):385–392, 1966.
[15] K. Sirlantzis, M. C. Fairhurst, and M. S. Hoque. Genetic algorithm for multiple classifier system configuration: A case study in character recognition. In J. Kittler and F. Roli, editors, *Lecture Notes in Computer Science (LNCS 2096) - Multiple Classifier Systems*, pages 99–108. Springer, 2001.
[16] Image Processing and Computer Vision Research Group, Department of Electronic Engineering, University of Kent, Canterbuty CT2 7NT, UK.

0-7695-1692-0/02 $17.00 © 2002 IEEE