gradient algorithm is applied on the same training set using different values for γ^0, and only the best error probability is retained. The error probability, averaged over the four training sets, is reported in Fig. 1.

 20 September 1995
Electronics Letters Online No: 19951412

C. Diamantini (*Istituto di Informatica, Dipartimento di Elettronica, Università di Ancona, Via Brecce Bianche, I-60131 Ancona, Italy*)

A. Spalvieri (*Dipartimento di Elettronica e Informazione, Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133, Italy*)

References

1 KOHONEN, T., BARNA, G., and CHRISLEY, R.: 'Statistical pattern recognition with neural networks: Benchmarking studies'. Proc. IEEE Int. Conf. on Neural Networks, San Diego, CA, July 1988, Vol. 1, pp. 61–68

2 KOHONEN, T.: 'The self organizing map', *Proc. IEEE,* 1990, **78**, (9), pp. 1464–1480

3 FUKUNAGA, K.: 'Introduction to statistical pattern recognition' (Academic Press, New York, 1972)

4 DIAMANTINI, C.: 'Quantizzazione vettoriale adattativa con applicazione al riconoscimento statistico di pattern'. PhD Thesis, University of Ancona, 1995

5 VAN TREES, H.L.: 'Detection, estimation, and modulation theory, Part III' (John Wiley & Sons, New York, 1971), Chap. 2

6 MCLEAN, G.F.: 'Vector quantization for texture classification', *IEEE Trans.,* 1993, **SMC-23**, pp. 637–649

Recognition of chain-coded handwritten character images with scanning *n*-tuple method

S. Lucas and A. Amiri

Indexing terms: Character recognition, Handwriting recognition

A method of applying n-tuple recognition techniques to handwritten OCR, which involves scanning an n-tuple classifier over a chain-code of the image, is described. The traditional advantages of n-tuple recognition, i.e. training and recognition speed, are retained, while offering superior recognition accuracy, as demonstrated by results on three widely used data sets.

Introduction: There is currently much interest in high-performance OCR for off-line applications such as the processing of handwritten forms, and on-line applications such as user-interfaces in pen-based personal organisers.

N-tuple systems are commonly used in OCR systems where speed is important. They have the advantage over most other techniques of offering both extremely fast training and extremely fast recognition even when implemented in software. Furthermore, should greater speed be necessary, they have simple hardware implementations. The only disadvantage is that recognition accuracy (on many problems, including OCR) tends to be not quite as good as more computationally intensive methods, such as multi-layer perceptron neural networks, or statistical/structural feature-based methods [1].

Here we present a new OCR method. Each character is first captured as a binary image and then chain-coded [2] to convert it to a 1-D string of symbols. The details of this procedure are not important; suffice to say that all the important information in the image i.e. the edges left by the trace of the pen, is now encoded in the chain-code string. A modified n-tuple system called the scanning n-tuple (sn-tuple) is then used to model and recognise these chain-code strings.

N-tuple classifiers and scanning n-tuple: In standard n-tuple classifiers [3] the d-dimensional (discrete) input space is sampled by m n-tuples. The range of each dimension in the general case is the alphabet $\Sigma = \{0, ..., \sigma-1\}$ but most n-tuple methods reported in the literature are defined over a binary input space where $\sigma = 2$ and $\Sigma = \{0, 1\}$.

Each n-tuple defines a fixed set of locations in the input space. Let the set of locations defining the jth n-tuple be $n_j = \{a_{j1}, a_{j2}, ..., a_{jn} | 1 \leq a_{ji} \leq d\}$ where each a_{ji} is chosen as a random integer in the specified range. This mapping is normally the same across all classes. For a given d-dimensional input pattern $\mathbf{x} = x(1) \ldots x(d)$ an address $b_j(\mathbf{x})$ may be calculated for each n-tuple mapping n_j as shown in eqn. 1.

$$b_j(\mathbf{x}) = \sum_{k=1}^{n} x(a_{jk}) \times \sigma^{k-1} \qquad (1)$$

These addresses are used to access memory elements, where there is a memory n_{cj} for each class c in the set of all classes C and n-tuple mapping n_j. We denote the value at location b in memory n_{cj} as $n_{cj}[b]$. The set of all memory values for all the n-tuple mappings for a given class we denote M_c, the model for a class c. The size of the address space of each memory n_{cj} is σ^n.

During training the value at location $n_{cj}[b]$ is incremented each time a pattern of class c addresses location b. During recognition, there are three established ways of interpreting the value at each address: binary, frequency weighted and probabilistic [4]. Only the probabilistic version will be derived here for the case of the scanning n-tuple, but results are also quoted below for the standard binary n-tuple classifier.

The 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 input space is now a variable length string $\mathbf{y}$ of length $\|\mathbf{y}\|$ rather than a fixed-length array.

We redefine sn-tuple address computation as follows for offset o relative to the start of the string as shown in eqn. 2

$$b_{jo}(\mathbf{y}) = \sum_{k=1}^{n} y(o + a_{jk}) \times \sigma^{k-1} \qquad (2)$$

The probability that this address is accessed by a pattern from class c is given in eqn. 3 where $N_{cj} = \Sigma_{b=0}^{\sigma^n-1} n_{cj}[b]$.

$$P(b_{jo}(\mathbf{y})|M_{cj}) = \frac{n_{cj}[b_{jo}(\mathbf{y})]}{N_{cj}} \qquad (3)$$

From this we calculate the probability of the whole string given sn-tuple model M_{cj}, under the assumption that the n-tuples at different offsets in the string are statistically independent:

$$P(\mathbf{y}|M_{cj}) = \prod_{o=1}^{\|\mathbf{y}\| - \max(a_{jk} \forall k \in \{1,\ldots,n\})} P(b_{jo}(\mathbf{y})|M_{cj}) \qquad (4)$$

As pointed out by Rohwer [4], the assumption of statistical independence between n-tuples is unrealistic, but there exists as yet no superior alternative. Note that eqn. 4 is very similar to the equation for the probability of a sequence given the statistical n-gram language models used in speech recognition [5], except that now we are able to model long-range as well as short range correlations. The probability of a string $\mathbf{y}$ given all the sn-tuple models of class c is given in eqn. 5.

$$P(\mathbf{y}|M_c) = \prod_{j=1}^{m} P(\mathbf{y}|M_{cj}) \qquad (5)$$

Subsequent pattern classification proceeds according to Bayes' theorem. If the prior class probabilities are assumed to be equal, as they are here, then the maximum likelihood decision is to assign the pattern to the class c for which $P(\mathbf{y}|M_c)$ is a maximum.

The algorithm for training the scanning n-tuple is as follows: all the memory contents are initialised to zero. For each pattern (string) of each class and for each mapping n_j, we scan the mapping along the string $\mathbf{y}$ from beginning to end by adjusting an offset o. In each case we increment the value at memory $n_{cj}[b_{jo}(\mathbf{y})]$ where the address calculation $b_{jo}(\mathbf{y})$ is defined in eqn. 2.

Although the sn-tuple introduces a new loop in the training and recognition stages (iteration over all offsets o) optimal results with the sn-tuple method can be achieved with just a few sn-tuples. For example, the results shown below use just 4 sn-tuples, compared to 40 n-tuples. Each architecture achieves a training rate of about 2000 characters per second and a recognition rate of about 200 characters per second on a SUN Classic rated at 55MIPs.

Results: Experimental results are reported for three widely used (and readily available) databases of handwritten digits: Essex, CEDAR and Concordia. In the original data sets, the images vary greatly in size, and the position of the character within a bitmap may also vary. To work well, *n*-tuple methods require the image to first be size and position normalised. For these experiments, a bounding box for each original character image was automatically computed; the image within this box was then scaled to fit on a 16 × 20 grid and then submitted directly to the *n*-tuple classifier, or chain-coded and then fed to the *sn*-tuple classifier. Best results for the standard binary *n*-tuple system were obtained using 40 8-tuples. Best results for the *sn*-tuple were obtained using 4 5-tuples, with evenly spaced offsets, spaced 2, 3, 4 and 5 places apart, respectively.

Table 1 shows the test set recognition accuracy for a standard binary *n*-tuple system, and a probabilistic *sn*-tuple system. Note that the *sn*-tuple achieves superior performance in every case.

Table 1: Data set sizes and recognition rates on test sets

Method	Essex	CEDAR	Concordia
	%	%	%
Training	3,900	19,000	15659
Testing	1,800	2,300	2000
n-tuple	90.6	91.8	89.6
sn-tuple	91.4	97.6	92.8

To give some idea of statistical significance, further experiments were conducted based on a random sampling methodology. For each entry in Table 2, 10 experiments were performed. For each experiment, a training set and disjoint test set of 500 characters per class for each set were created by randomly sampling the CEDAR training set. The results show the recognition accuracy obtained when applying both the binary and probabilistic *n*-tuple systems both to the image or, as *sn*-tuples, to a chain-code of the image. The advantages of adopting both a probabilistic interpretation and a scanning mode of application are clear. The reason the results are poorer than those for the CEDAR set in Table 1 is that fewer samples were used for training (500 against ~2000 per class). We also tested a conventional *n*-gram model [5] using this random sampling methodology on the same sample set. Best results obtained were 92.5% with $n = 6$. The *sn*-tuple gives superior performance because it is able to model longer-range constraints owing to its use of non-consecutive offsets.

Table 2: Results of 10 experiments (mean with standard deviation in brackets) comparing performance of binary *n*-tuple and probabilistic *n*-tuple applied to image, and scanned across chain-code (string)

	Fixed image	Scanning (chain-code)
Bin	86.0% (0.37)	92.1% (0.41)
Prob	62.5% (0.27)	69.5% (0.24)

Conclusions: *N*-tuple recognition methods have long been established as extremely fast and robust, while offering good performance. The *sn*-tuple as applied here retains these advantages while offering superior accuracy. The recognition rate of 97.6% at 200 characters ps (cps) on the CEDAR data compares well with a range of other techniques recently quoted [1], where the fastest method scored 96.1% at 66cps and the most accurate method scored 98.9% at 10cps.

Acknowledgments: This work was partially supported by EPSRC grant GR/J 52969, EPSRC grant GR/J 66959 under the DTI/EPSRC Speech and Language Technology (SALT) scheme.

29 September 1995

Electronics Letters Online No: 19951438

S. Lucas and A. Amiri (*Department of Electronic Systems Engineering, University of Essex, Colchester CO4 3SQ, United Kingdom*)

References

1 FAVATA, J., SRIKANTAN, G., and SRIHARI, S.: 'Handprinted character/digit recognition using a multiple feature/resolution philosophy'. Proc. Fourth Int. Workshop Frontiers in Handwriting Recognition, 1994, pp. 57–66

2 LUCAS, S.: 'High performance OCR with syntactic neural networks'. Proc. IEE 4th Int. Conf. Artificial Neural Networks, London, 1995, pp. 133–138

3 ALEKSANDER, I., and STONHAM, T.: 'Guide to pattern recognition using random-access memories', *IEE Proc. Comput. Digit. Tech.*, 1979, **2**, pp. 29–40

4 ROHWER, R., and MORCINIEC, M.: 'The theoretical and experimental status of the n-tuple classifier'. Tech. Report NCrG/4347: Neural Computing Research Group, Aston University, UK, 1985

5 JELINEK, F., MERCER, R., and BAHL, L.: 'The development of an experimental discrete dictation recognizer', *Proc. IEEE*, 1985, **73**, pp. 1616–1624

MMIC internal electric field mapping with submicrometre spatial resolution and gigahertz bandwidth by means of high frequency scanning force microscope testing

A. Leyk and E. Kubalek

Indexing terms: MMIC, Scanning probe microscopy

High frequency scanning force microscope (HFSFM) testing enables, among other measurements, the probing of device internal electric potential distributions within monolithic microwave integrated circuits (MMICs) with both high spatial and temporal resolution. Two planar components of the electric field can be calculated from this data and their local distribution can be mapped. For the first time 2-D distributions of electric field components on an interdigital capacitor within an MMIC at frequencies up to 6 GHz were evaluated from HFSFM potential measurements, demonstrating the submicrometre spatial resolution and gigahertz bandwidth of this new MMIC internal electric field mapping technique

Introduction: The rising complexity and bandwidth of modern monolithic microwave integrated circuits (MMICs) requires advanced techniques for external and internal node testing and failure analysis [1]. A test system needs to have sufficient spatial resolution for submicrometre structural dimensions, and a simultaneously sufficient bandwidth for gigahertz signals [2]. Promising solutions to meet these requirements have been reported, using new scanning force microscope based test systems [4 – 7]. Improvements led to a high frequency scanning force microscope (HFSFM) test system, presently capable of measuring electric potentials up to 100GHz [3]. MMIC internal electric field mapping by electro-optic testing proved helpful for failure analysis [5, 6], but suffers from limited spatial resolution above 1μm. Direct comparisons with quantitative topography measurements of the tested MMIC are not possible. The use of an HFSFM test system overcomes these disadvantages and enables MMIC internal electric field mapping with submicrometre spatial resolution and gigahertz bandwidth in direct comparison with quantitative topographical micrographs of the tested MMIC, as demonstrated in this Letter.

System setup and operation modes: The HFSFM, depicted in Fig. 1, is centred on a commercially available SFM. The probe consists of a conducting atomically-sharp doped silicon tip fixed to a cantilever, and it can be moved at a constant distance from the MMIC surface by an xyz piezo-scanner. Owing to different distance-dependent tip-MMIC interactions, the tip is either attracted or repelled from the surface of the MMIC. This results in a bending of the cantilever, which is optically detected and analysed by a lock-in amplifier. In topographic mode, the distance between the tip and the MMIC surface is within a few nanometres, so the van der Waals interaction becomes dominant, enabling quantitative