ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

Novel Modular Weightless Neural Architectures for Biometrics-based Recognition

Konstantinos Sirlantzis[1], Gareth Howells[2] and Bogdan Gherman[3]

1, 2, 3 - University of Kent, Canterbury UK - Dept of Electronics
Department of Electronics, University of Kent, Canterbury, Kent, CT2 7NT - United Kingdom
E-mail: {k.sirlantzis, w.g.j.howells, bbg3}@kent.ac.uk

Abstract. We introduce a novel weightless artificial neural architecture based on multiple classifier systems. In this, different modules of a network specialise in recognising specific classes of a multiclass recognition task. Each of these modules comprises individual RAM addresses which store frequency-based probabilistic estimates of how likely it is to observe this pattern as a feature of the training examples available from a particular class. The class-wise likelihood of observing a combination of addresses for each class is calculated as a sum-based scheme (one of the most commonly used multi-classifier fusion methods). The classification decision is finally obtained by choosing the class with the highest pseudo-posterior probability for an address combination. Tests of our system on a face recognition problem using Minchinton cell encoding for mapping regions of interest (ROIs) to the network's input layer showed very encouraging results.

Keywords: Multi-classifier systems, Weightless Neural Network, Fast learning.

1 Introduction

Modern multiple classifier systems are being increasingly employed in practical application domains where the required performance level exceeds that achievable from a single pattern classifier or expert. These systems typically employ concurrently a number of distinct classifiers, each of whose defining characteristics are able to address an aspect of the pattern classification task as a part of the overall problem domain in question. Face recognition is a prominent example of difficult realistic problems, and has long attracted the application of a wide range of powerful classification methods [7], including variations of multi-classifier systems [2].

This paper introduces a weightless neural network as a multi-classifier system composed of simple base classifiers which are applied to greyscale face images. The system possesses the following significant properties: a) fast learning, so that the significance of class distinguishing properties are immediately realised by the system; b) problem domain independence, i.e., unlike traditional neural architectures, it may be employed for any practical pattern classification task as it is not limited by the number of pattern classes present within the problem domain. In addition, the system retains the design simplicity associated with weightless neural architectures. The proposed technique employs a type of weightless system using a pseudo-posterior (frequency-based) calculation mechanism to assimilate the classification potential of each of the component class discriminants. The remainder of this paper is organized as follows: Section 2 introduces the network architecture. Section 3 describes the data and experimental setup and results, with concluding remarks in Section 4.

2 Network architecture

The structure of the Weightless Neural Network (WNN) was inspired by the early work of Bledsoe and Browning on the n-Tuple model [1] of WNN, the overall view is presented in Figure 1. Similar architectures have been reported recently to be considered for facial recognition tasks by De Souza et al. [6], but in contrast our approach performs an automated facial recognition without manual intervention on the input database.

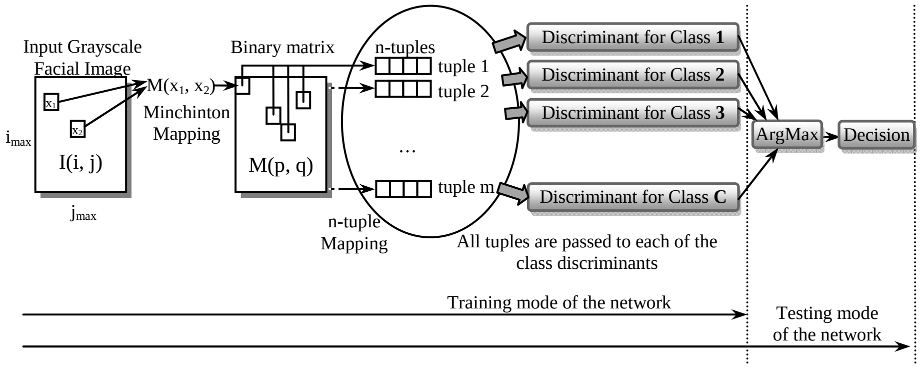

Fig. 1: Overall view of the WNN for a tuple size n=4, number of tuples m=2576

The implemented weightless network is making use of the following modules to achieve its goals, (see Figure 1): a) a greyscale to binary pre-processing module using the so called Minchinton cell encoding [3]; b) a mapping of the resulting binary matrix to a series of n-bit RAM addresses (tuples); c) an ensemble of class discriminant modules, one for each class to be classified by the network, each comprising a number of RAM address ranges; d) a fusion module that uses the output of the class discriminants to reach a final decision.

Each greyscale image is pre-processed using a Minchinton cell mapping of Type 0 as described in [3] and [4]. The employed Minchinton mapping has the following definition:

$$M(x_1, x_2) = \begin{cases} 1, & if\ I(x_1) > I(x_2) \\ 0, & otherwise \end{cases}$$

where I[x] represents one greyscale pixel value in the input image I at location x, respectively x_1 and x_2 are random coordinate values. If $I[x_1] > I[x_2]$ then the resulting binary pixel value is 1 otherwise it is 0. This mapping is applied for every pixel of the input image, without taking a particular pixel location into account twice, and thus covering the entire image. We have conducted experiments with the image coverage varying from 10% to 100% in 10% increments. The reduced coverage is achieved simply by not taking into account all the randomly selected pairs of pixels.

By applying the Minchinton mapping to the input image a binary matrix is obtained which is half the size of the original image. *n* number of bits are then extracted in order to create the *n*-tuples, the order of which can be random or in the order in which they appear in the binary matrix. It is not important which of these two methods of choosing the tuples is used, a hypothesis which is backed up by experiments. Both the Minchinton and n-tuple mappings are kept invariant for the whole lifetime of the network that is, for training and testing alike.

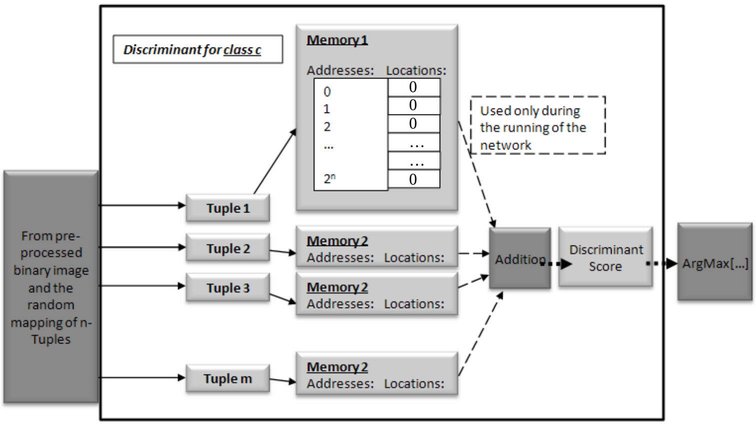

Fig. 2: Class discriminant in detail.

The main component of the WNN is the ensemble of class discriminants (Figure 2). A discriminant consists of a series of address spaces similar to Random Access Memory (RAM) components that are attached to each of the n-tuples, where the value of the n-tuple is employed as the address of the RAM location, being incremented when the network is trained. This way the network is counting the frequency of a bit pattern sequence encountered in the binary matrix. In training mode, the network counts the occurrence of binary input patterns from the binary matrix representing the transformed greyscale input image for all the tuples and in each of the discriminants for all the classes separately. In testing mode, an unknown face image is mapped to a binary matrix and all the RAM locations that are addressed by all of tuples are summed together, in each class discriminant separately. This produces the score of each class discriminant, assigning the final class label to the discriminant with the highest score.

3 Experimental setup and Results

To test our system we used the AT&T database [5] formerly "The ORL Database of Faces" due to its wide acceptability (see [7] for a very recent report). The database is composed of 400 images, each having 92x112 pixels resolution, and encoded in PGM 8 bit gray scale format. The pictures are taken from 40 individuals having 10 pictures of each person with a variable posture, facial expression and different occlusions of the face, e.g. hair, glasses or beards. The total number of 400 pictures was split into two sets, training and testing. The training set consisted of 240 images (60% of the database) while the testing set was comprised of 160 images (40% of the database).

We used two types of random data splitting techniques, one which didn't take into account classes and resulted into having different class prior probabilities for each class (i.e. the number of images in the training sample differ from class to class), and the second type of data splitting was based on class membership and thus produced equal prior probabilities for each class (i.e. equal number of images in the training set for all classes) referenced as "**Not equal class priors**" and "**Equal class priors**" respectively. The first type of data splitting builds a somewhat more realistic scenario since this is often the situation in real world applications whilst the second case is the one more often presented in related literature.

Our experiments aimed to investigate the effect of tuple size with respect to the two different class priors presented above, as well as, the effect of tuple size with respect to different levels (percentages of face image area) the network input mapping was covering. To this end we run two sets of experiments. In the first, the input mapping covers 100% of the facial image areas but the number of images per class used for training (representing class priors), as well as the tuple size is varied. In the second, the number of training images per class is fixed but the percentage of image area covered when establishing the networks input mapping is varied as the tuple size is varied, see the results in Tables 2 and 3.

We repeated all our experiments 10 times to obtain statistical estimates of the performance of the proposed systems in each one of the different sets of experiments described previously. The result tables show the average performance on the **testing set**s (recognition rates) obtained from each set of training-testing cycles. During each of these cycles the corresponding network is recreated in order to cancel out the effects of particular image pixel to network input mappings (i.e. the repetitive use of specific locations of the image which may have stronger or weaker than average discriminatory properties and which may bias the average performance observed).

To put our investigation in an implementation-related context we include in Table 1 the tuple sizes used in our experiments with their corresponding a) number of tuples required for 100% coverage of the facial image (row 1), b) number of bits required by each RAM, and c) memory size required by each class-wise discriminant.

	Tuple size				
	28	**23**	**16**	**8**	**4**
Number of tuples	368	448	644	1,288	2,576
Size of one RAM	$2^{28} \cdot 368$	$2^{23} \cdot 448$	$2^{16} \cdot 644$	$2^{8} \cdot 1288$	$2^{4} \cdot 2576$
Memory used by one discriminant	92 Gb	4 Gb	40 Mb	322 Kb	40 Kb

Table 1: Comparison of memory requirement of different tuple sizes.

Table 2 shows the recognition rates achieved by both using unequal and equal class priors during training. The smaller the tuple size the greatest the effect of class prior is. A starting difference of approximately 47% when a 4-tuple is used, then reduced to only around 4% when we use a 23-tuple. Performance increased as we increase the tuple size up to a point and then starts decreasing again. However, it appears that the tuple sizes producing the maximum performance are not the same for equal and unequal class priors in the training set. The best performance overall is achieved using training data where classes have equal number of examples.

	Tuple size				
	4	**8**	**16**	**23**	**28**
Not equal class priors	**38.13%**	**65.75%**	**71.06%**	**79.75%**	**68.94%**
Equal class priors	**85.31%**	**88.02%**	**90.19%**	**83.94%**	**74.06%**

Table 2: Face recognition accuracy averaged over 10 runs.

Table 3 presents a much more complex situation. While for each tuple size (columns) performance can be seen to increase with increasing image coverage of the networks input mapping (rows), as it is expected, for the same level of coverage recognition rates appear to increase as the size of tuple is decreasing.

	Tuple size			
	28	**23**	**16**	**8**
10% coverage	**35.88%**	**51.19%**	**73.13%**	**84.88%**
20% coverage	**46.06%**	**64.75%**	**79.94%**	**86.81%**
80% coverage	**71.94%**	**82.56%**	**88.44%**	**88.25%**
90% coverage	**73.25%**	**84.94%**	**89.56%**	**87.38%**

Table 3: Recognition accuracy at varying image coverage and tuple sizes.

The latter a somewhat counterintuitive observation which, however, given that in these experiments we kept class priors equal and fixed, seems to indicate the importance of the number of tuples used as a factor which strongly affects performance of these type of systems. Also, while the best performance in the majority of cases is observed with a 16-tuple as suggested by results of Table 2 as well for the equal priors scenario, we can also observe that reasonably high recognition rates can be achieved by a lower image coverage level (and as such much lower memory and computational load)

Finally, in Table 4 we present performance levels achieved on the same database as reported in [2]. These indicate that the simple and non-optimised systems investigated in this work appear to outperform much more complex models such as those based on Hidden Markov Models and Principal Component Analysis (PCA).

At the same time considerable difference of performance (approx. 7%) can be seen to have been achieved by optimising the choice of discriminant modules to be fused in the context of n-tuple-based multiple classifier systems (see rows 4 and 5 in Table 4). The currently presented work is also developed within the same context, which strongly suggests that intelligent optimisation of tuple-based weightless network systems offers a fruitful avenue for further future investigation.

Classification algorithm recognition accuracy rates	
Top-bottom Hidden Markov Model [2]	87.00%
Eigenfaces (PCA-based) using Euclidean Distance [2]	90.00%
Best system reported in this work (16-tuple, equal priors)	90.19%
Bit-plane decomposition-based n-tuple: Optimised Combination of Ordered Layers [2]	97.10%
Bit-plane decomposition-based n-tuple: Optimised Combination of Random Layers [2]	93.30%

Table 4: Comparison of recognition accuracy results of other recognition algorithms as reported by Sirlantzis, Hoque and Fairhurst [2].

4 Conclusions

This paper presented an initial investigation of the performance properties of a novel weightless network architecture used for face recognition. Examined under two different scenarios of equal and unequal class priors in the training dataset our results suggest a number of parameters (such as the number of tuples used in conjunction with the area of the image covered by the network's input mapping) which require attention and possibly optimisation. Results have been seen to be comparable and in a number of cases better than alternative methods reported in the literature.

References

[1] W. W. Bledsoe, I. Browning, Pattern recognition and reading by machine, *Eastern Joint IRE-AIEE-ACM computer conference*, pages 225-232, December 1-3, Boston, Massachusetts, 1959.

[2] K. Sirlantzis, S. Hoque, M.C. Fairhurst, Classifier Diversity Estimation in a Multiclassifier Face Recognition System based on Binary Feature Quantisation, Proceedings of the *4th International Conference on Recent Advances in Soft Computing* (RASC 2002), A. Lotfi, J. Garibaldi and R. John, eds., Nottingham, UK, pp. 84-90, 2002 December

[3] J.M. Bishop, P.R. Minchinton and R.J.Mitchell, Real time invariant grey level image processing using digital neural networks, Proc *IMechE Conf, EuroTech Direct '91 – Computers in the engineering industry*, pages 187-199, Birmingham 1991.

[4] James Austin, editor. *RAM-Based Neural Networks*. World Scientific Publishing, 1998.

[5] Harter, F. Samaria and A. "Parameterisation of a stochastic model for human face identification." *2nd IEEE Workshop on Applications of Computer Vision*, December 1994. (Credit for the AT&T / ORL database)

[6] A. F. De Souza, C. Badue, F. Pedroni, E. Oliveira, S. S. Dias, H. Oliveira, and S. F. de Souza. Face Recognition with VG-RAM Weightless Neural Networks. In Proceedings of the *18th International Conference on Artificial Neural Networks* (ICANN'08), pages 951-960, Prague, Czech Republic, September 2008.

[7] Licheng Jiao, Rui Hu, Weida Zhou and Yi Gao, Face Recognition Using Kernel Uncorrelated Discriminant Analysis, Lecture Notes in Computer Science, Advances in Multimedia Modeling, Volume 4352/2006, pages 415-422, Springer-Verlag, 2006