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.

FPGA-based enhanced probabilistic convergent weightless Network for human Iris recognition

P. Lorrentz[1], W. G. J. Howells[2], and K. D. McDonald-Maier[3]

[1]*Department of Electronics, University of Kent, Canterbury, Kent, CT2 7NT, UK Tel. 0044(0)7813089916, Fax: +441227 456084. e-mail: plorrentz@gmail.com*
[2]*Department of Electronics, University of Kent, Canterbury, Kent, CT2 7NT, UK .Tel: +44 122723724. Fax: +441227 456084, e-mail: W.G.Howells@kent.ac.uk*
[3]*Department of Computer Science, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK. Tel. 0044(0)1206874376, e-mail: kdm@essex.ac.uk*

Abstract. This paper investigates how human identification and identity verification can be performed by the application of an FPGA based weightless neural network, entitled the Enhanced Probabilistic Convergent Neural Network (EPCN), to the iris biometric modality. The human iris is processed for feature vectors which will be employed for formation of connectivity, during learning and subsequent recognition. The pre-processing of the iris, prior to EPCN training, is very minimal.

Structural modifications were also made to the Random Access Memory (RAM) based neural network which enhances its robustness when applied in real-time.

1. Introduction

The properties of human eye iris are stable throughout the life of an individual, and therefore make the iris a suitable biometric modality. The biometric properties of every iris have additionally been shown to be unique [1][2][3]. Efficient and effective methods by Daugmann [4][5][6][7] consists of application of 2-D Gabor wavelet transformation followed by Hamming distance measures. Liam [8] employs a trained self-organizing neural network in eye iris recognition. Also Wang [9] utilises RBF and Fisher (discriminant analysis) classifier for classification of iris.

In this paper, a Weightless Random Access Memory (RAM) based neural network called EPCN [10] will be employed. The area of application of RAM based neural networks is limited [11], more so in biometric identification of iris, due to their limited ability to generalise, and problems associated with input pre-processing. But FPGA-based weightless neural networks have much capability to minimise noise than any other alternatives, which implies more efficiency and reliability. This has been demonstrated also in [11] where RAM-based neural networks are applied to image processing, fingerprint identification, etc. The advantages of employment of FPGA implemented RAM based EPCN however includes speed, and portability [10]. Some of this advantages are explained in section 2 where EPCN is introduces, while experiments and results are presented in section 3. Conclusions and possible areas of future research are indicated in section 4.

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.

2. An FPGA based Enhanced Probabilistic Convergent Neural (EPCN) Network

The EPCN belongs to a group of neural network known as RAM based neural network, or n-tuple classifier. The hardware architecture of EPCN is schematically shown in Figure 1.

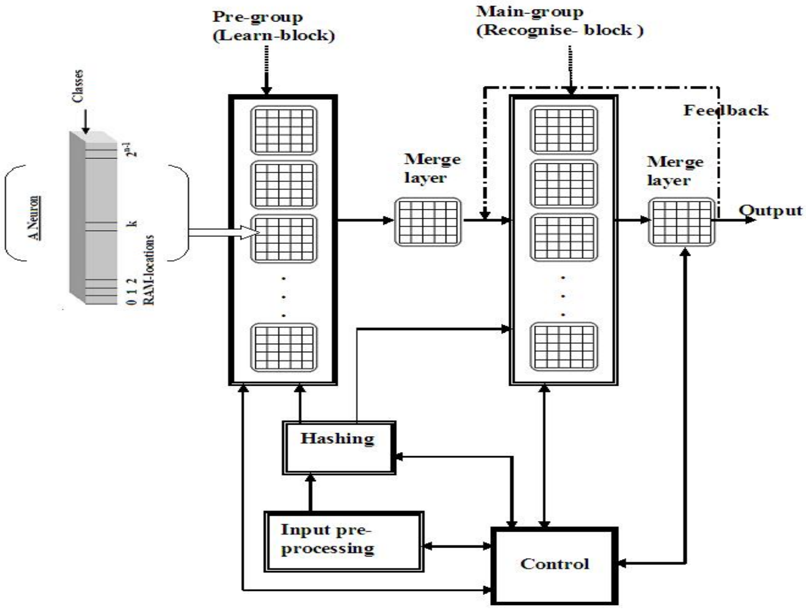

Figure 1: A simplified schematic of Hardware Architecture of EPCN network.

The pre-processing of iris patterns is performed within the input pre-processing block, and the addresses for RAM locations within a neuron are formed from an input pattern by a hashing function in the Hashing block. The hashing function is derived from XOR and Maximum-length Shift-register code (MLSR) [12]. The MLSR make use of a parity polynomials h(p) given by

$$h(p) = p^k + h_{k-1}p^{k-1} + \ldots + h_0p^0$$

Hashing occurs prior to both learning and recognition. Learning utilises the pre-group layers (learn-block) each of which consist of neurons, as shown in figure 1. Each neuron's RAM-locations are independently assessed by classes comprising the database. The frequency of asses by a class to a RAM-location is recorded in that location. When a neuron's RAM-locations are normalised, the result is the probability of occurrence. Secondly, the decision (output) of EPCN in a recognition task is a probability of occurrence of classes which is sometimes called confidence level (value) – thus the term "probabilistic" in the naming of EPCN. Recognition utilises the main-group layers (recognise-block) each of which consist of neurons. Reading from and/or writing to the neurons are mediated by connectivity, the size of which is determined by the number of address-line known as *n-tuple*. Values in these neurons are usually averaged and adjusted by a specifiable positive whole number known as *division*. Learning is initialized by the control block, and a learn-complete signal is sent to the control block when learning ends. The recognition procedure is treated similarly. There is a feedback (see figure 1) of the output to the main-group. After several iterations, the same group of values may be obtained at the output. The neural network is said to converge to a solution (example of which is shown in figure 2) to the task – thus the naming "convergence" in EPCN. Where there is no convergence,

```
grp.desout =

{{01011010} {00110000} {00010100} {00110010} {00000000} {00000000} {00101000} {00000000}
{00000110} {00000000} {00110010} {00001000} {00110010} {00000000} {00000110} {00000000}
{00000000} {00011000} {00000000} {00000000} {00001000} {00110010} {00000000} {00000000}
{00110010} {00001100} {00110010} {00110010} {00000010} {00000000} {00110010} {00001000}
{00000000} {00000010} {00000000} {00010000} {00000000} {00110010} {00000000} {00001000}
{00000100} {00000100} {00000010} {00000000} {00000010} {00000000} {00000000} {00000000}
{00001100} {00000000} {00000000} {00000100} {00000000} {00000000} {00000110} {00000000}
{00000000} {00000000} {00110010} {00010000} {00110010} {00110010} {00000000} {00000000}}
```

Figure 2 An example output of EPCN where each binary value (in brackets) indicates the relative probability of each of the 64 classes of the iris.

the solution oscillates between series of values. Since the output of EPCN is expressed in probability values, it is said to *converge in probability* to the solution.

The input pre-processing block designed here is distinct and substantially different from that in [10]. This is because collective properties representative of irises are much different from that representative of handwritten numbers. Iris databases share common attributes, and there are over 200 of them. It is the value of these attributes that identifies an iris in EPCN, and not their presence/absence. This implies that a different pre-processing algorithm may not be required by every iris database. The pre-processing algorithm and other particularities are now presented.

2.1 Pre-processing

The following pre-processing steps were experimentally selected against other alternatives. The EPCN initiates pre-processing whereby the outer circular edge, and the inner circle of the eye separating the iris from the pupil, are identified and emphasized by employing the Canny operator. This is followed by wavelet decomposition, and then binary coding. Important steps are noted below:-

- Wavelet compression is applied to each eye iris (e.g. Figure 3(a)) pattern so as to remove feature redundancies, and decrease the size of the image.
- Gaussian filtration and Canny edge detector are applied to emphasize the circular edges of the pupil and the iris.

(a) (b)

Figure 3: Example of (a) the image of an eye iris before pre-processing. (b) The pattern derived from (a) by pre-processing, which is used for learning (or subsequent recognition) of EPCN network.

- This is followed by wavelet decomposition and binary coding. The binary coding give rise to Figure 3(b), which is a suitable form accepted by EPCN network.

2.2 Adaptation of EPCN to Hardware

The EPCN is implemented on Virtex-II Pro by using VHDL in Xilinx. The neural network is designed entirely on-chip. On-chip learning, and subsequent recognition, offers some flexibility of number precision, increased independence, and speed. This is in comparison to off-chip learning, and chip-in-the-loop learning. The mapping of this neural network onto FPGA is governed by the constraints which include accuracy, space, and speed. Most of the processes within EPCN architecture runs in parallel, especially the division and multiplication algorithms. While the busses and data paths are pipelined for concurrent utilisation. These pipelining and parallel strategies lead to a very compact structure of EPCN, thus saving space. All mathematical processes are either Boolean or discrete. Therefore the speed of execution of the network's processes is relative to the amount of data at input to the system. This leads to overall increase in speed of the neural network, and reduce power consumption.

Most other weightless neural networks, e.g. WISARD, utilises a unit known as pyramid which is a discriminator. The WISARD architecture [11] contains several discriminators while EPCN contains layers. Each pyramid of a discriminator contains only 0's and 1's, while each neuron of layers of EPCN contains natural numbers. Each pyramid of WISARD in recognition mode contains either a 1 or a 0 at its apex. Collection of these 1's and 0's identifies an object. In contrast, EPCN converge in probability to a solution. Methods of forming connectivity in EPCN may be applied to WISARD and vice versa.

3. Experimentation

The source of database employed in this experiment is [13][20]. The irises were scanned by TOPCON TRC50IA optical device connected with SONY DXC-950P 3CCD camera [14]. There are 64 participants, which naturally provide 64 classes for the neural network. A couple-eye refers to both the left- and right-eye of an individual. The EPCN network is trained with 64 classes, each consisting of the 3 instances of a couple-eye iris. This makes 6 irises per class for training. All irises comprising the database were used during the recognition phase. The number of iris per class is decreased in step of 1, while all irises are used for recognition. Figure 4 shows the errors, in percentage (%), recorded while decreasing the number of iris/class used in training EPCN.

The learning phase corresponds to registration (or enrollement) of an individual. Cases of enrolment failure were not observed. During the classification phase, each iris is presented to EPCN. This corresponds to identification or verification of an individual. Ambiguous cases contribute to the percentage of unclassified pattern, which in turn gives the False Rejection Rate (FRR). Similarly, an incorrect classification contributes to the False Acceptance Rate (FAR).

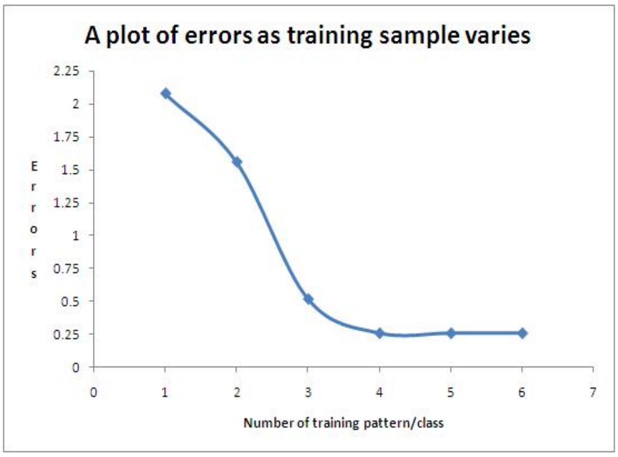

Figure 4: Summary of results obtained by verying the number of pattern per class used during learning.

From figure 4, there is a sharp increase in error rate, when the number of training pattern was decreased from 3 downto 1, implying that different instances of the iris is necessary for training the network. It is not possible, in this work, to trace every source of errors, but it is possible to quantify total sum of errors, and this is shown in the vertical axis of figure 4. The error trend shows that from 6 to 3 iris/class is sufficient for good recognition. When the number of pattern is decreased from 6 down to 4, consistent performance is obtained. These indicate the ability of EPCN to generalise well. Table 1 shows the comparison of FPGA-based EPCN with some other neural networks.

Methods/Database	Performance (%)	Platform/Circuit size	Time/Cycles	
p-RAM (on 4 printed digits)	93%	IBM PC 66MHz	6200 sec.	[15]
EPCN (on handwritten numbers 0 - 9)	80%	occupied	7.5e-8 sec	[10]
EPCN optimised (on Eye Iris)	99.17%	Virtex-II Pro.; 2402 SLICEs used	7.0e-8 sec	
Information Theory (Mutual information, on Eye Iris)	99.05%	Mathlab 6.5	-	[14]
Back-propagation	-	Xilinx XC4005XL-PQ; 196 CLB used.	1879 ns	[16]
Space Vector Modulation (on Neural network)	max. Saving 83.22%	Xilinx XCV50hq240 FPGA, max. 1096 SLICEs	max. 149 cycles	[17]
Support Vector Machine (on SONAR, Diabetes, etc)	max. 94.4%	Spartan3 XC3s50pq208-5; max 1244SLICEs	-	[18]
Radial Basis Function (on pressure sensor)	97.20%	Virtex-II FPGA; 14334 SLICEs used	206 ms	[19]

Table 1: Comparison of EPCN with other Neural networks implemented on other platforms.

In [14], 99.05% accuracy was obtained using mutual information. While in this work, 99.17%, on average, is obtainable from a single EPCN. The reason could be inferred from the fact that EPCN learns about a class independently, and that each iris has about 266 measurable features, [4][5][6][7]. This is usually sufficient to uniquely distinguish any iris. Comparisons on table 1 in absolute numerical terms are to be treated with scepticism. The order of magnitude seems more significant, since the aims, objectives, experimental setup, and results formation varies widely. From table 1, it seems the Virtex-II family of FPGA would accommodate more complex design as compared to other types FPGA and at greater speed also.

4. Conclusion

Results obtained indicate that the presented pre-processing method is very efficient and welcomes further enhancement, such as the inclusion of iris detection and localisation algorithm. The presented pre-processing algorithm may be capable of processing almost all iris images, but any iris detection and localisation algorithm employed may depend on the hardware used for image acquisition.

A hardware implementation of a weightless neural network has been developed and employed successfully in an iris recognition system. The performance of the system has been shown to be potentially suitable and effective for integration within a security system that requires identity verification.

References

[1.] Adler F.H.: Physiology of the Eye. Clinical Applications., C.V. Mosby, London 1965.

[2.] Davson H.: Davson's Physiology of the Eye. Macmillan, London 1990.

[3.] Wildes R.P.: Iris recognition: An Emerging Biometric Technology, Proceedings of the IEEE, vol. 85, No. 9, pp. 1348 – 1363, September 1997.

[4.] Daugman J.G.: Statistical richness of visual phase information: Update on recognizing persons by iris patterns, Int. J. Comput. Vis. 45 (2001) 25 – 38.

[5.] Daugman J.G.: The importance of being random: Statistical principles of iris recognition. Pattern recognit. 36 (2003) 279 – 291.

[6.] Daugman J.G.: Biometric Personal Identification System Based on Iris Analysis U.S. Patent No. 5,291,560 U.S. Governments, Washington 1994.

[7.] Daugman J.G.: High confidence visual recognition of person of statistical independence. IEEE Trans., Pattern Anal. Intell. 15(1993) 1148 – 1161

[8.] Liam L.W, Chekima A., Fan L.C., Dargham J.A.: Iris recognition using self-organizing neural network. In: Proc. Student conference on Research and Development, Shah Alam, Malaysia, 169 – 172, 2002.

[9.] Wang Y., Tan T. Jain A.K: Combining Face and Iris Biometrics for Identity Verification, AVBPA 2003, LNCS 2688, pp.805-813, Springer-Verlag Berlin Heildelberg 2003.

[10.] Lorrentz P., Howells W.G.J, McDonald-Maier K.D.: An Analysis of Hardware Configurations for an Adaptive Weightless Neural Network, WCE'08, July 2008.

[11.] Austin J.: RAM-based neural networks, *World Scientific*, 1998.

[12.] John G. Proakis: Digital Communication, 4^{th} Edition, McGraw Hill Int. Edition, 2001.

[13.] Michal Dobeš and Libor Machala, Iris Database, http://www.inf.upol.cz/iris/

[14.] Dobeš M., Machala L., Tichavský P., Pospíšil J.: Human Eye Iris Recognition Using the Mutual Information. Optik Volume 115, No.9, p.399-405, Elsevier 2004, ISSN 0030-4026.

[15.] Ouslim M., Curtis M.: Pattern recognition based on a probabilistic RAM net using n-tuple input mapping, IEE Proc.-Vis. Image Signal Process., Vol. 145, No. 6, Dec. 1998.

[16.] Coric S., Latinovic I., Pavasovic A.: A Neural Network FPGA Implementation: Issues & Applications, NEUREL-2000, Yugoslavia, sept. 25-27, 2000

[17.] Muthuramalingam A., Himavathi S., Srinivasan E.: Neural Network Implementation using FPGA, Int. Journal of Information technology Voln. 4, Nr. 2, 2007.

[18.] Fasial M. Khan, Mark G. Arnold, and William M. Pottenger: Hardware-based Support Vector Machine Classification in Logarithmic Number System, IEEE Symposium on Circuit and Systems (ISCAS), May 2005.

[19.] Jagdish C. Patra, Theodora A. Devi, and Pramod K. Meher: radial Basis Function Implementation of Intelligent Pressure Sensor on Field Programmable Gate Arrays, ICICS, 2007.

[20.] Dobeš, M., Martinek J., Skoupil D., Dobešová Z., Pospíšil J.: Human eye localization using the modified Hough transform. Optik, Volume 117, No.10, p.468-473, Elsevier 2006, ISSN 0030-4026, 2006.