Evolving Wavelet Neural Networks for Breast Cancer Classification

Maryam Mahsal Khan **Stephan K. Chalup** **Alexandre Mendes**

School of Electrical Engineering and Computer Science
The University of Newcastle,
Callaghan NSW 2308, Australia
Email: MaryamMahsal.Khan@uon.edu.au
Email: Stephan.Chalup@newcastle.edu.au
Email: Alexandre.Mendes@newcastle.edu.au

Abstract

Digital Mammograms are x-ray images of the breast and one of the preferred early detection methods for breast cancer. However, mammograms are still difficult to interpret, and associated with this problem is a high percentage of unnecessary biopsies, misdiagnoses and late detections.

The focus of this research is to use neuroevolutionary mechanisms for detecting breast cancer from mammographic images. The aim is to design a sophisticated classification tool that detects breast cancer at its early stages, so that treatment has a better chance of success.

Wavelet neural networks have the ability to capture and extract information at various frequency levels by using different dilation and scaling values of the wavelet function. In this work, the wavelet neural network parameters are evolved using on the concept of Cartesian Genetic Programming, resulting in an evolved neural network which is trained for mass diagnosis.

In the reported study the proposed algorithm achieves a classification accuracy of 89.57% on a real dataset composed of 200 images. Such a computer-based classification system has the potential to provide a second opinion to the radiologists, thus assisting them to diagnose the malignancy of breast cancer more precisely.

Keywords: Breast Cancer, Mammography, Cartesian Genetic Programming, Evolution, Neural Networks, Wavelet Neural Networks, Neuroevolution

1 Introduction

Breast cancer is the second leading cause of cancer-related deaths in Australian women, accounting for 15.5% of them. It is estimated that one in eight Australian women will be diagnosed with the disease before the age of 85 (*Breast Cancer Care WA; accessed June* 2014).

Digital mammograms are digital x-ray images of the breast; and regularly used for cancer screening. It is one of the earliest and most reliable detection methods, with cancer being indicated by the presence of a microcalcification - calcium deposit within the breast tissue or masses. The identification and assessment of potentially cancerous areas is a tedious, time-consuming and challenging task, which requires specialised expertise. Such assessments might also lead to misdiagnosis, which is why Computer Aided Detection (CAD) systems provide a valuable second opinion for the classification of suspicious areas as cancerous (malignant) or non-cancerous (benign).

Artificial Neural Networks (ANN) are a computational model represented by simulated neurons (called units) with weighted connections between them. There are a number of evolutionary algorithms devised in the past decade with different strategies of evolving either connection weights, network topology, or both (this last case known as TWEANN – Topology and Weight Evolving Artificial Neural Networks). An important example of such evolving networks is the NEAT (Neuroevolution of Augmented Topology), proposed by (Stanley & Miikkulainen 2002). The algorithm has the capability to evolve both structures and weights depending on the complexity of the problem, and is not dependent on a predefined network structure. Also the recently proposed neuroevolutionary algorithm namely ANN evolved via Cartesian Genetic Programming (CGPANN) has also been applied on different domains of engineering with success (Khan et al. 2013).

Standard classifiers like Support Vector Machines and Multilayer Perceptron, despite their success in many domains, display some limitation on varying complex tasks, e.g. intelligent control, language learning, etc (Byun & Lee 2002, Muhlenbein 1990). Wavelets - referred to as a 'microscope' in mathematics (Cao et al. 1995), act as high compression nodes that represent non-linearities effectively (Fang & Chow 2006). Wavelet neural network has been applied on a variety of problems with great success, e.g. time-series analysis and prediction (Cao et al. 1995, Chen et al. 2006), signal de-noising (Zhang 2007), classification and compression (Kadambe & Srinivasan 2006, Subasi et al. 2005), density estimation (Hasiewicz 1997), non-linear modelling (Billings & Wei 2005), etc. Cartesian Genetic programming has been used particularly in digital circuit optimization(Miller & Thomson 2000). In this research the wavelet neural network parameters are evolved using the Cartesian Genetic Programming so that the evolved wavelet neural network benefits from the strength of wavelets and overcome the limitations of standard classifiers; thus possibly contributing to classification, prediction and control problems.

Our research focuses on the development of a novel neuroevolutionary algorithm not only to classify mass abnormalities identified in digital mammograms; assessing its potential to be a part of a high-confidence CAD system for diagnosis, but also to exploit it further in the intelligent control domain. For this reason the potential of the algorithm is first tested on a well-

Copyright ©2014, Australian Computer Society, Inc. This paper appeared at Australasian Data Mining Conference (AusDM 2014), Brisbane, 27-28 November 2014. Conferences in Research and Practice in Information Technology, Vol. 158. Richi Nayak, Xue Li, Lin Liu, Kok-Leong Ong, Yanchang Zhao, Paul Kennedy Eds. Reproduction for academic, not-for profit purposes permitted provided this text is included

researched and a reasonably challenging dataset, that has been used by many researchers (McLeod & Verma 2013*b*, Zhang et al. 2010, Verma et al. 2009*a*). The dataset is also tested on two existing neuroevolutionary algorithms, namely ANN evolved via Cartesian Genetic Programming and Neuroevolution in Augmented Topology.

The remainder of the paper is divided into five sections. Section 2 describes Cartesian Genetic Programming (CGP) and the proposed algorithm - Wavelet Neural Networks evolved via CGP. Section 3 highlights the dataset for cancer detection and the literature review surrounding that database. Section 4 describes the methodology, followed by analysis and discussion of the results. Finally, Section 6 concludes the paper.

2 Background

2.1 Cartesian Genetic Programming

Cartesian Genetic Programming (CGP) (Miller & Thomson 2000) is an evolutionary programming technique used particularly for digital circuits optimisation. CGP genotypes are of finite length and have an integer representation, where genes represent nodes and each node corresponds to sets of input genes and a function. The input genes can be a program input, or outputs of other nodes. Functions are either logical or arithmetical e.g. AND, OR, addition, subtraction, etc. The output of the genotype is either a node output or the program input itself.

There are two basic types of evolution strategies (μ, λ)-ES and $(\mu + \lambda)$-ES (Beyer & Schwefel 2002). μ represents the number of parent population and λ refers to the number of offspring produced in a generation. In (μ, λ) offspring replaces the parent as the fittest is selected from λ, while in $(\mu + \lambda)$-ES the fittest is selected from both parents and offspring for the next generation. Cartesian Genetic programming uses the $(1 + \lambda)$ strategy, where $\lambda = 4$, is commonly adopted. i.e. a single parent is mutated based on a mutation rate 'τ' to produce 4 offsprings.

Figure 1(a) is an example of a finite length genotype with two inputs (x_0, x_1) and one output, where the encircled number represents the output. It represents a 2×2 architecture i.e. it has 2 rows and 2 columns. The number of inputs to each node is 2. The functions used are the logical OR and AND gates, displayed as f_0 and f_1, respectively. Figure 1(b) is the graphical representation of the genotype. The graph represents active and inactive nodes. Inactive nodes are nodes that do not participate in the output computational process (shown in light grey). Based on the output x_5, the phenotype of the corresponding genotype is shown in Figure 1(c).

2.2 Cartesian Genetic Programming Wavelet Neural Network (CGPWNN)

Wavelet Neural Networks have three layers, namely input, hidden and output layers. The input layer represents the input to the network. The hidden layer consists of wavelet neurons ψ, known as wavelons, with scaling α and translation β parameters, as shown in Figure 2(a). Therefore, the input presented to the wavelons are scaled and translated, which transforms the input pattern. The output layer approximates, or sums the input coming from the hidden layer. Each output from the hidden layer is multiplied by a weight w_i, where i corresponds to the wavelet neuron index. There are a number of wavelet functions ψ that can be used, e.g. Gaussian derivative, Mexican hat, Morelet, Haar, etc. The choice of wavelet used in the application depends on the system itself. The structure of a single hidden layer wavelet network is displayed in Figure 3. The output of the network is mathematically expressed in Eq. (1):

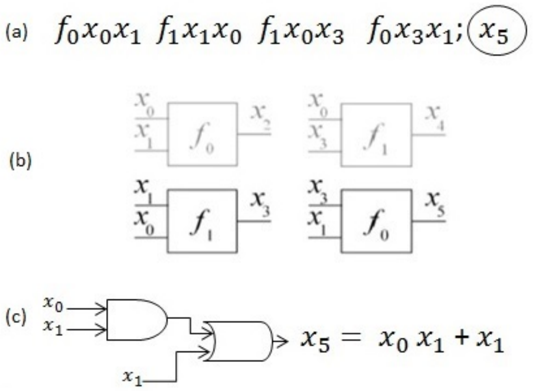

Figure 1: (a) An example of a 2×2 CGP genotype with 2 inputs and 1 output. (b) Graphical representation of the genotype in (a). (c) Phenotype corresponding to the genotype in (a).

$$y(x) = \theta + \sum_{i=1}^{m} w_i \psi_i(x) + \sum_{j=1}^{n} c_j x_j \qquad (1)$$

where θ is the bias to the output neuron and $c_j x_j$ represents the direct connection of input to the output representing a linear model (Alexandridis & Zapranis 2013).

The CGP representation has been used to evolve artificial neural networks previously (Khan et al. 2013). Similarly, in the current paper we will use the CGP representation, but to evolve wavelet neural networks. A node in CGP corresponds to a wavelet neuron in CGPWNN. Figure 2(b) shows a wavelon of CGPWNN. The genes that make up a wavelon include: input x_{ij}, connection c_{ij}, motherwavelet ψ, translation β and scale α where x_{ij} = [1, number of program inputs], c_{ij} = {0, 1}, ψ = [1, number of wavelet functions], β = [0, 1] and α = [0, 1] respectively. The input and connection genes occur in pairs, i.e. if the input to a wavelon is 2, then it would constitute two inputs and two connection genes.

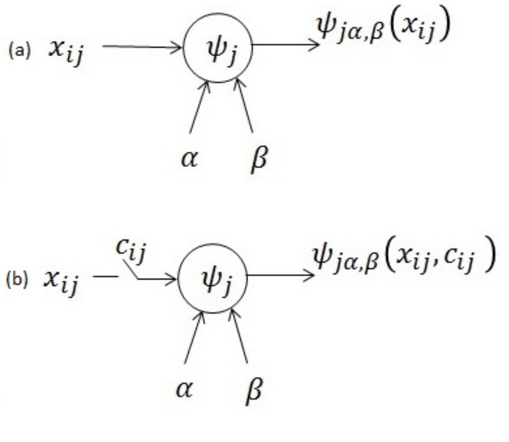

Figure 2: (a) Diagram of a Standard Wavelet Neuron. (b) CGPWNN-Wavelet Neuron.

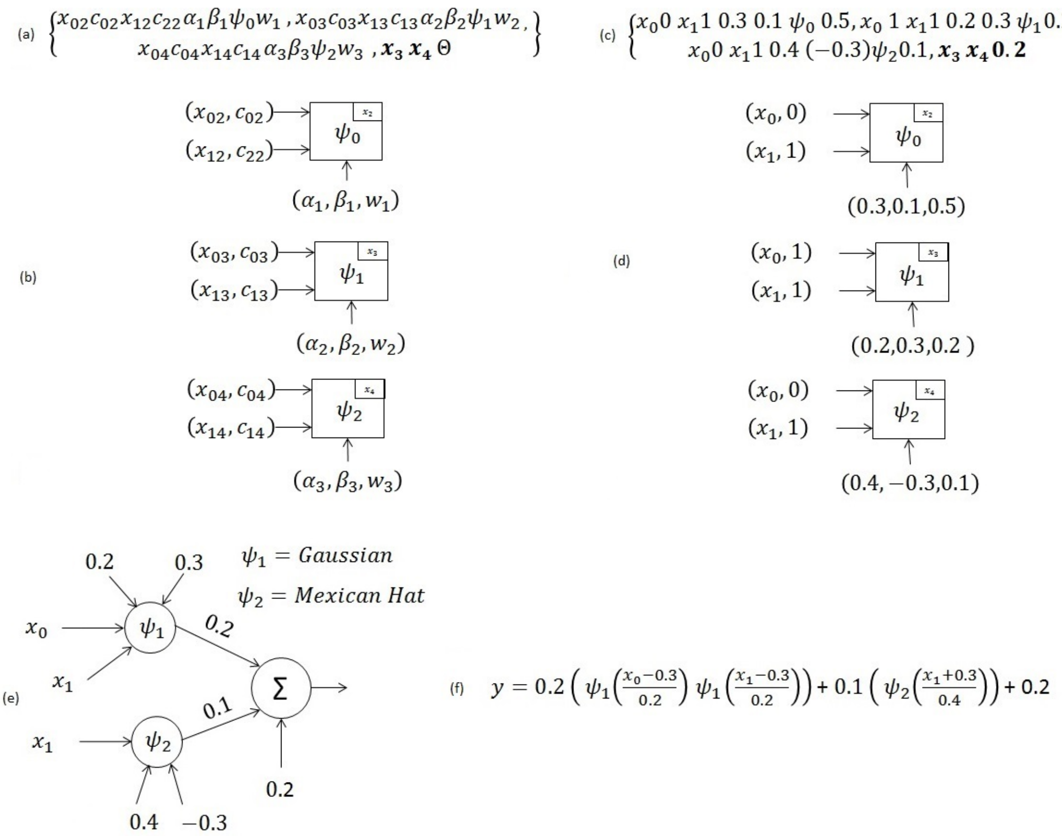
(a) $\begin{Bmatrix} x_{02}c_{02}x_{12}c_{22}\alpha_1\beta_1\psi_0w_1, x_{03}c_{03}x_{13}c_{13}\alpha_2\beta_2\psi_1w_2, \\ x_{04}c_{04}x_{14}c_{14}\alpha_3\beta_3\psi_2w_3, \boldsymbol{x_3\ x_4}\ \Theta \end{Bmatrix}$
(c) $\begin{Bmatrix} x_0 0\ x_1 1\ 0.3\ 0.1\ \psi_0\ 0.5, x_0\ 1\ x_1 1\ 0.2\ 0.3\ \psi_1 0.2, \\ x_0 0\ x_1 1\ 0.4\ (-0.3)\psi_2 0.1, \boldsymbol{x_3\ x_4\ 0.2} \end{Bmatrix}$
(b)
(x_{02}, c_{02})
(x_{12}, c_{22})
ψ_0
x_2
(α_1, β_1, w_1)
(x_{03}, c_{03})
(x_{13}, c_{13})
ψ_1
x_3
(α_2, β_2, w_2)
(x_{04}, c_{04})
(x_{14}, c_{14})
ψ_2
x_4
(α_3, β_3, w_3)
(d)
$(x_0, 0)$
$(x_1, 1)$
(0.3,0.1,0.5)
$(x_0, 1)$
$(x_1, 1)$
(0.2,0.3,0.2)
$(x_0, 0)$
$(x_1, 1)$
(0.4, −0.3,0.1)
(e)
0.2
0.3
$\psi_1 = Gaussian$
$\psi_2 = Mexican\ Hat$
x_0
x_1
x_1
ψ_1
ψ_2
0.2
0.1
Σ
0.2
0.4
−0.3
(f) $y = 0.2\left(\psi_1\left(\frac{x_0-0.3}{0.2}\right)\psi_1\left(\frac{x_1-0.3}{0.2}\right)\right) + 0.1\left(\psi_2\left(\frac{x_1+0.3}{0.4}\right)\right) + 0.2$

Figure 4: (a) A 3 × 1 example genotype of CGPWNN. (b) Graphical representation of the genotype. (c) Random values assigned to (a). (d) Graphical representation of the genotype with assigned random values from (c). (e) Phenotype of (d). (f) Mathematical representation of phenotype.

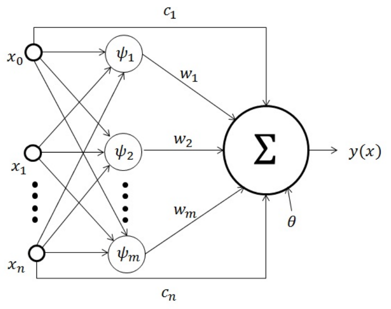

Figure 3: Structure of a Wavelet Neural Network.

Figure 4(a) is an example of a 3 × 1 CGPWNN genotype with two inputs: x_0 and x_1. The number of inputs to each wavelet neuron is 2. The number of outputs from the network is also 2, i.e. x_3 and x_4. The wavelet functions used are Gaussian, Mexican hat and Haar, labelled as ψ_1, ψ_2 and ψ_3, respectively. The bias gene is incorporated at the end of the genotype, and marked as θ. Figure 4(b) is the graphical representation of such genotype. The number at the top right corner of each wavelet neuron corresponds to the node index. Suppose that we assign random values to the genes, as shown in Figure 4(c). In that case, its graphical representation is displayed in Figure 4(d). The phenotype of the assigned genotype is Figure 4(e), and can be mathematically expressed as in Figure 4(f).

3 Case study: Mass classification

Classifying suspicious areas in digital mammograms is a crucial, difficult problem, and one of the significant processes for the early detection of breast cancer. In the current paper we are investigating one of the challenging and publically available benchmark datasets for breast cancer diagnosis using evolutionary neural networks. In this section we shall discuss in detail the features used for mass classification and the associated literature survey.

3.1 Database and features utilized

The Digital Database for Screening Mammography (DDSM), from the University of South Florida, is an online repository of mammographic images collected from different hospitals, with different resolutions (*Digital Database for Screening Mammography; accessed May* 2014, Heath et al. 1998, 2001). The suspicious regions are manually marked on the film by two experienced radiologists. These regions are represented as chain codes which can be easily extracted from the image file for further analysis.

A total of 25 features are extracted from the regions marked on the mammographic images scanned on the HOWTEK scanner, at 43.5 micron per pixel spatial resolution (Kumar, Zhang, & Verma 2006). The feature set includes 18 grey level features, based on the grey level pixel values of suspicious areas using the statistical formulas shown in Table 1 (where T = total number of pixels; g = index value of image I; k = number of grey levels (4096); $I(g)$ = grey level of pixel g in image I; and $P(g)$ is the probability of the grey level g occurring in image I. Also, each case in DDSM contains information specified by an expert radiologist using BIRADS (Breast Imaging Reporting and Data System) lexicon. The four BIRADS attributes are *density*, *mass shape*, *mass margin* and *assessment*. *Patient age* and *subtlety* values are also extracted from each mammographic record, while *calcification association* is added by (Kumar, Zhang & Verma 2006). Those 7 features are human interpreted. The list of features along with their descriptions is shown in Table 1.

3.2 Literature Survey

There are many research papers surrounding breast cancer diagnosis and classification. The following literature focuses on research that used the dataset features reported above, for the purpose of comparing results. In 2005, (Zhang, Kumar & Verma 2005) proposed a hybrid classifier that used statistical classifiers (Logistic Regression (LR) and Discriminant Analysis (DA)) output probabilities as second order features, combined in a feature set with other 14 grey level and 6 human extracted features. The modified feature set was then tested on several classifiers, including neural networks, and genetic neural networks, with 3 random splits of the dataset. A maximum accuracy of 91% for the LRDA-GNN classifier was obtained. Furthermore, in their papers, (Zhang & Kumar 2006) statistically analyzed the various features using SPSS, and 4 key features (assessment, age, margin and shape) were identified. They were then used in conjunction with neural networks and decision trees (CART) (Breiman et al. 1984, Steinberg & Colla 1997, Steinberg & Golovnya 2006) and C5.0 (Quinlan 1993, RuleQuest-Research 2014) for classification purposes. Accuracy was higher in comparison to using the whole feature subset, which meant that feature extraction improved the performance of the classification. Also, they proposed that using Logistic Regression alone on the 7 human extracted features attains high classification accuracy, with an AUC (area under curve) of 0.979 (Zhang et al. 2010).

(Panchal & Verma 2006) exploited different feature subsets in terms of its classification accuracy using auto associative and classifier neural networks. A total of 14 grey level, 4 BIRADS, plus patient age and subtlety features were selected, and subsequently divided into six feature subsets. The main objective behind the research was to identify key features in breast cancer detection. The study determined that grey level and BIRADS features perform better, with a training accuracy of 100% and testing accuracy of 92%. Training and testing was done using a 50/50 data split.

(Kumar, Zhang, & Verma 2006) used decision trees (CART and C5.0) at different cost ratios for mass classification on the whole feature set. Data was also split 50/50 for training and testing. Results showed a maximum of 91% accuracy, for a cost ratio of 1:1 using CART; at the cost of a higher standard deviation.

Verma's series of works introduced a number of algorithms for classifying the mass dataset (Verma 2008, Verma et al. 2009*b*,*a*). In one of those, an additional neuron in the hidden layer was proposed (based on the number of classes), improving both the memorization and generalization ability of the network. A different training mechanism for the additional neurons was also devised. Based on that approach, the training and testing accuracy improved to 100% and 94% respectively, where the classifier was trained and tested on a 50/50 data split using 6 of the human extracted features. Also, Verma introduced two soft cluster-based neural networks, where the clusters were formed within a neural network layer i.e. SCBDL (soft cluster based direct learning) & SCNN (soft cluster neural network). By using 10-fold cross validation with both algorithms, a maximum of 94% in SCNN and 95% in SCBDL accuracy was achieved. A comparison with other clustering algorithms (SVM, K-means and SOM) showed accuracies of 86.5%, 84.5% and 76%, respectively.

In 2011, (McLeod & Verma 2011) proposed a multi-cluster support vector machine for mass classification. The K-means algorithm was used to generate the clusters for benign and malignant classes. The resulting clusters were then used for classification on a standard SVM. The MCSVM obtained an average accuracy of 94.5%, while standard SVMs reached 87.5% using 10-fold cross validation and 6 human extracted features. McLeod also observed an approximate increase of 3% in accuracy using the same cluster based approach on other classifiers, namely radial basis function networks and multilayer perceptrons (McLeod & Verma 2010). The accuracy of the classifiers were improved further by using neural network ensemble classifiers in (McLeod & Verma 2012*a*,*b*, 2013*a*,*b*). The networks in the ensemble varied the number of neurons in the hidden layer, between 2 to 150 neurons. The maximum number of classifiers in the ensemble was limited to 40 in (McLeod & Verma 2012*b*), and 202 in (McLeod & Verma 2013*a*). A 10-fold cross validation was used for testing the methods. The final ensemble network was composed of 127 classifiers, which attained an accuracy of 99%. It is noteworthy that in order to classify 200 data rows a total of 127 classifiers was needed. Similar improvement was observed in an LCA ensemble (94%), compared to LCA (87%) alone in (Pour et al. 2012).

3.3 Training and testing sets

A total of 200 suspicious areas were manually extracted from the Digital Database for Screening Mammography dataset. Half of those areas represented benign tumours and the other half was malignant (Zhang, Kumar & Verma 2005).

3.3.1 Training on 70% of the data

The first part of the experiment involved training and testing the classifiers by splitting the dataset into 70% and 30%, respectively, with equal contribution of benign and malignant samples to each group. Similar to Verma's and McLeod's studies, 6 human-interpreted features were used in all of our experiments (breast density, mass shape, mass margin, assessment, subtlety and patient age).

3.3.2 10-fold cross validation

The second part of the simulation incorporated a 10-fold cross validation strategy for testing the classifiers. In that case, the dataset is divided into 10 subsets, where 9 subsets are combined into a training set, and the remaining subset is used as the test set. This is repeated 10 times, always with a different selection of subset for testing, and the average accuracy is reported.

In this work, the classifier's output is thresholded, based on a threshold value of $\theta = 0$, to classify the samples as benign (0) or malignant (1). That is mathematically expressed by Eq. (2).

Table 1: Features and description of the Digital Database for the Screening Mammography (DDSM) dataset. (Zhang, Verma & Kumar 2005)

Features	**Description**
Grey Level Features	
Minimum Grey Level	Minimum grey level in the suspicious area
Maximum Grey Level	Maximum grey level in the suspicious area
Perimeter of Suspicious Area	Count of pixels at the boundary of the extracted area
Mean Boundary Grey Level	BAG = Average Grey Level at the boundaries
Number of Pixels	Count of pixels in extracted area
Mean Histogram	$AHg = (1/k) \sum_{j=0}^{k-1} N(j)/T$
Energy	$Egy = \sum_{g=0}^{k-1} [P(g)]^2$
Entropy	$Etp = -\sum_{g=0}^{k-1} P(g) log_2[P(g)]$
Standard Deviation	$\sigma = \sqrt{\sum_{g=0}^{T-1} (g - AG)^2 P(g)}$
Skew	$Skw = (1/(\sigma_j)^3) \sum_{g=0}^{k-1} (g - AG)^3 P(g)$
Modified Energy	$MEgy = \sum_{g=0}^{T-1} [P(I(g))]^2$
Modified Entropy	$MEtp = -\sum_{g=0}^{T-1} P(g) log_2[P(I(g))]$
Modified Standard Deviation	$m\sigma = \sqrt{\sum_{g=0}^{T-1} (I(g) - AG)^2 P(I(g))}$
Modified Skew	$MSkw = (1/\sigma_j^3) \sum_{g=0}^{T-1} (I(g) - AG)^3 P(I(g))$
Kurtosis	$Kur = (1/(\sigma_j)^4) \sum_{g=0}^{k-1} (g - AG)^4 P(g)$
Mean Grey Level	$AG = 1/T \sum_{g=0}^{T-1} I(g)$
Difference	$Dff = AG - BAG$
Contrast	$Ctr = Dff/(AG + BAG)$
Human Interpreted Features - BIRADS	
Breast Density	Density of breast tissue; rated 1-4
Abnormality Assessment Rank	Seriousness of abnormality; rated 1-5
Mass Shape	Morphological descriptor, e.g. round, oval, lobulated, irregular etc.; rated 1-9
Mass Margin	Morphological descriptor, e.g. circumscribed, microlobulated, obscured etc.; rated 1-5
Human Interpreted Features - Others	
Subtlety	Subjective abnormality measure; rated 1-5
Patient Age	Age of patient at the time of mammography
Calcification Association	Relation of mass to calcification; categorized as yes or no

$$ClassifierOutput = \begin{cases} 0, \text{if } Output \geq \theta \\ 1, \text{if } Output < \theta \end{cases} \tag{2}$$

3.3.3 Performance measures

The performance of the classifiers is evaluated based on the following metrics:

1. **Training Accuracy** (Tr_{Acc}): fraction of correctly trained samples.

2. **Testing Accuracy** (Te_{Acc}): fraction of correctly classified samples as expressed in Eq. (3), also known as the classification accuracy. The higher the percentage, the better is the classifier performance.

$$Te_{Acc} = \frac{(TP + TN)}{P + N} \tag{3}$$

where TP represents true positive cases, i.e. accurate classification of benign samples; TN represents true negative cases, i.e. accurate classification of malignant samples; and $(P + N)$ is the total number of positive and negative test samples.

3. **Sensitivity** (*Sens*): measurement of the fraction of true positive cases, mathematically represented in Eq. (4):

$$Sens = \frac{TP}{(TP + FN)} \tag{4}$$

where FN is the number of false negatives – *Type 2* error – where the classification of a malignant case as benign is a severe mistake.

4. **Specificity** (*Spec*) Statistical measurement of the fraction of true negative cases mathematically represented as in Eq.(5):

Table 2: Performance of CGPANN with 70/30 split between training and testing samples. The best configuration is with [1 × 100], $I_E = 3$ and $O_p = 4$, indicated in bold. That configuration was then tested using 10-fold cross-validation (bottom row).

Configuration			Accuracy %				Active Parameters	
Structure	I_E	O_p	Tr_{Acc}	$Te_{Acc}(\sigma)$	Sens	Spec	Neurons(%)	Features
1 × 50	3	2	91.85	87.05(2.87)	83.31	91.73	18.13	4.53
		4	92.83	87.83(3.28)	85.06	91.07	21.60	4.57
	6	2	92.67	87.67(2.56)	85.02	90.74	32.33	5.70
		4	**93.57**	87.72(3.63)	87.10	88.36	32.87	5.70
1 × 100	3	4	92.59	**89.11**(2.84)	88.45	92.00	18.83	4.87
		8	93.14	88.22(3.46)	87.63	88.82	25.90	5.23
	6	4	92.90	87.61(2.90)	85.51	89.96	28.03	5.97
		8	92.52	87.73(3.58)	86.28	89.30	43.63	6.00
Best configuration - 10-fold cross-validation								
1 × 100	3	4	92.57	87.15(5.24)	86.10	88.91	16.92	5.01

Table 3: Performance of CGPWNN & CGPWNN with linearity disabled on 70% Training and 30% Testing Dataset. The best configuration indicated in bold was then tested using 10-fold cross-validation.

Configuration			Accuracy %				Active Parameters	
Structure	I_E	O_p	Tr_{Acc}	$Te_{Acc}(\sigma)$	Sens	Spec	Wavelons(%)	Features
CGPWNN								
50 × 1	3	4	94.02	89.16(2.86)	84.00	96.19	7.93	4.33
		8	93.47	87.67(3.32)	82.65	94.48	15.87	4.97
		12	94.07	87.94(2.94)	83.57	93.61	23.33	5.67
	6	4	93.02	88.27(3.48)	83.87	94.00	7.93	4.83
		8	92.07	87.27(5.73)	82.16	94.31	15.60	5.87
		12	93.78	88.61(3.16)	85.71	92.01	23.93	6.00
100 × 1	3	4	93.76	89.50(3.17)	84.61	95.98	3.97	4.10
		8	93.45	88.27(3.23)	83.60	94.45	7.90	5.17
		12	93.95	88.16(2.76)	83.64	94.09	11.87	5.60
	6	4	93.80	**89.57**(2.85)	84.90	95.64	4.00	5.00
		8	**94.14**	88.11(3.06)	83.96	93.41	8.00	5.97
		12	93.97	88.61(1.82)	84.10	94.49	12.00	6.00
Best configuration - 10-fold cross-validation								
100 × 1	6	4	92.99	88.60(4.83)	86.84	91.14	3.99	4.94
CGPWNN with linearity disabled								
50 × 1	3	4	93.59	88.22(2.94)	84.60	92.67	8.00	3.93
		8	94.30	**89.57**(3.64)	85.38	94.38	16.00	4.97
		12	94.28	88.83(2.89)	85.41	92.98	24.00	5.60
	6	4	93.47	87.89(2.68)	84.23	92.41	8.00	4.80
		8	93.61	89.11(2.30)	85.12	94.11	16.00	5.70
		12	93.83	88.89(3.25)	85.35	93.20	24.00	5.97
100 × 1	3	4	93.78	89.22(2.60)	85.22	94.23	4.00	3.87
		8	**94.59**	88.27(3.23)	84.69	92.68	8.00	5.23
		12	94.26	89.33(2.41)	85.40	94.25	12.00	5.60
	6	4	93.59	88.67(2.63)	83.72	95.31	4.00	4.73
		8	93.97	88.72(2.18)	84.53	94.05	8.00	5.87
		12	93.52	88.67(2.83)	84.31	94.27	12.00	5.97
Best configuration - 10-fold cross-validation								
50 × 1	3	8	94.09	88.03(5.36)	86.70	89.92	16.00	5.08

$$Spec = \frac{TN}{(TN + FP)} \tag{5}$$

where FP is the number of false positives – *Type 1* error – corresponding to the classification of a benign sample as malignant.

4 Experimental setup

4.1 CGPANN parameters

The first part of the experiment involves the evolution of genotypes under two random architectures [1 × 50] and [1 × 100], where rows = 1 and columns = 50 and 100, respectively, and with different parameter settings. The intent of having a single row is to have a fully connected feedforward network - a standard CGP configuration. The number of inputs to each neuron I_E were 3 and 6 and the number of outputs O_p were 2, 4 and 8, respectively. A $(1+9)$-ES, with $\lambda = 9$, and a mutation rate of 0.1% was used in all of the simulations, similar to (Khan et al. 2013). Each network was evolved for 50,000 generations. The activation functions used were sigmoid and hyperbolic tangent. Table 2 shows the different configuration of parameters for the network and their performance, using a 70/30 split between training and testing samples. The table shows the figures for training accuracy, testing accuracy along with the standard deviation of the accuracy of the 30 genotypes, sensitivity, specificity, active neurons and the number of selected features.

The results are averaged over 30 independent evolutionary runs. The best result was with [1 × 100], $I_E = 3$ and $O_p = 4$, with $Tr_{Acc} = 92.59$ and $Te_{Acc} = 89.11$. That configuration then proceeded for further testing, now using 10-fold cross-validation. Results indicate a training accuracy of 92.57% and a testing accuracy of 87.15%, using the average for 30 independent runs of the cross-validation. Sensitivity and specificity were 86.10% and 88.91%, respectively.

4.2 CGPWNN parameters

Similarly to CGPANN, two random CGPWNN architectures [50 × 1] and [100 × 1] were used. The number of columns was set to 1, as the number of hidden layers in a wavelet neural network is also 1. The number of inputs to each wavelon I_E was set at 3 and 6; and the number of outputs O_p were 4, 8 and 12. As in the previous case, a $(1+9)$-ES, with $\lambda = 9$, and a mutation rate of 0.1% was used in all of the simulations. Each network evolved for 50,000 generations. Wavelet functions used in the experiments were Gaussian, Mexican hat and Haar wavelets. The networks were trained on 70% of the data and tested on the remaining 30%, as before. The performance for each network configuration is shown in Table 3 and results represent the average of 30 independent evolutionary runs. Similar parameters were also used to train a modified version of CGPWNN - disabling direct connection of input to the output. The intent was to know whether input features modeled non-linearly would perform better; results are shown in the same table.

4.3 NEAT parameters

The main parameters used in NEAT for evolving neural network structure is shown in Table 4. The NEAT classifier was trained under both 70% training, 30% testing; and the 10-fold cross validation strategies. Table 5 shows the result of each training and testing set which is the average of 30 independent evolutionary runs.

5 Results and discussion

In Tables 2 and 3, there is no observable trend for accuracy as the networks' structure vary. Maximum training and testing accuracies achieved by CGPANN (from Table 2) are 93.57% and 89.11%. The maximum training accuracy of CGPWNN in Table 3 is 94.14% with a 100 × 1 structure with 6 inputs and 8 outputs. Analogously, the maximum testing accuracy was 89.57% with a network structure of 100 × 1 with 6 inputs and 4 outputs. By disabling the direct input connectivity to the output (see Table 3), the maximum training accuracy was 94.59% with a 100 × 1 network with 3 inputs and 8 outputs; and testing accuracy remained same i.e. 89.57% with a 50 × 1 structure with 3 inputs and 8 outputs. These results implies that the feature set can be modeled either way.

Table 4: NEAT algorithm parameters

Attribute	Value
Population size	150
Speciation (c1, c2, c3)	(1, 1, 0.4)
Crossover percentage	0.8
Mutation probability: Add node	0.03
Mutation probability: Add connection	0.05
Mutation probability: Recurrency	0.0
Mutation probability: Mutate weight	0.9

As mentioned in Section 4, the network with the maximum accuracy is used to train the data samples using 10-fold cross validation. Since 10-fold cross validation is a considerably more robust test strategy compared to training/testing split, we observed a relatively small performance decrease. CGPANN attained average testing accuracy of 87.15% (Table 2) while CGPWNN and its modified version (Table 3) obtained an average testing accuracy of 88.60% and 88.03%, respectively.

From Table 5, NEAT obtained training and testing accuracies of 90.59% and 89.11%, respectively. Both CGPANN and NEAT achieved the same accuracies, but the sensitivity of CGPANN was found to be 88.45% while that of NEAT was 86.67%. The 10-fold cross validation again resulted in a small decrease in the performance of the NEAT classifier to 84.63%. A similar reduction in classification accuracy was also observed in (McLeod & Verma 2011, Verma et al. 2009*a*).

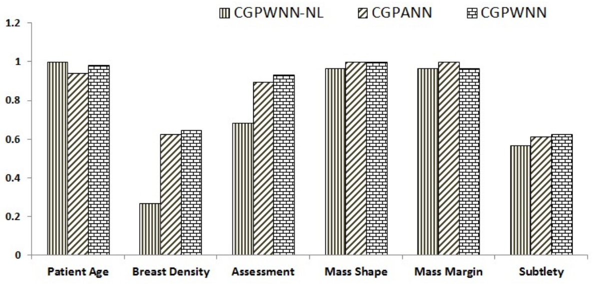

Figure 5: Feature selection of the evolved neural networks, where patient age, mass margin and mass shape were selected by the three algorithms all the time.

Figure 5 shows a histogram of the features selected using the networks (CGPWNN, CGPWNN-NL, CGPANN) with the best testing accuracy. The results are averaged over 30 independent evolutionary runs

Table 5: Performance of NEAT using 70/30 split and 10-fold cross validation strategies.

	Accuracy %			
Strategy	Tr_{Acc}	$Te_{Acc}(\sigma)$	*Sens*	*Spec*
70/30 split	90.59	89.11(3.02)	86.67	92.48
10-fold cross valid.	91.14	84.63(4.50)	85.55	85.93

Table 6: Performance of neuroevolutionary algorithms in terms of number of evaluations and computational time

Algorithm	Average number of evaluations	CPU time (in hours)
CGPANN	171,090	2.96
CGPWNN	179,670	1.35
CGPWNN-NL	154,160	1.30

and shown as percentages. In CGPANN and CGPWNN most of the genotypes selected four features: patient age, assessment, mass margin and mass shape – similar to (Zhang & Kumar 2006); breast density and subtlety were selected less often (approximately 60% of the time). Finally, CGPWNN - NL selected 3 features every time: patient age, mass margin and mass shape. Thus, one can argue that those are the most robust features from the features list, as they are selected by all methods all the time.

In Table 2, it is also observed that the maximum number of active neurons in the search space of 1×50 and 1×100 is 43.63%. Even though more than 57% of the genotype represents inactive genes or junk nodes, it has been shown that the presence of inactive genes actually is useful to the efficiency of the evolutionary process, due to the concept of neutrality (Miller & Smith 2006, Vassilev & Miller 2000, Yu & Miller 2001, 2002). Active nodes are only involved in the computational process which implies lesser delays. By increasing the number of outputs, the contribution of neurons from the pool of resources increases (Table 2).

In CGPANN, the computational process cannot be controlled via any genes (input, connection, weight, function, output, etc). On the other hand, in CGPWNN, since the architecture is forcibly $[n \times 1]$, the number of outputs ultimately controls the range of active wavelons in the search space. From Table 3, with increasing numbers of output nodes, the active wavelons in the search space increases, restricted to an upper limit of the total number of outputs; thus, forcing the evolutionary process to search for optimum solutions under a controlled computational environment.

A (1+9)-ES (used in our simulations) implies an evaluation of 10 genotypes in each generation. Table 6 shows the average number of evaluations for 30 independent evolutionary runs of CGPANN, CGPWNN and CGPWNN-NL genotypes with the maximum classification accuracy along with the average CPU time (in hours) to complete 50,000 generations. CGPWNN-NL has the lowest number of evaluations, at 154,160, as compared to CGPANN (171,090) and CGPWNN (179,670) and therefore, is the fastest learning algorithm among the three. For the average CPU time, the platform was a single core CPU @3.40GHz with Windows-XP 64-bit. We can clearly see that CGPANN took the longest time (2.96 hrs) in evaluating genotypes, while CGPWNN and CGPWNN-NL took 1.35 hrs and 1.30 hrs, respectively. That was somewhat expected, as in CGPWNN computational delay is controlled via outputs; hence the time required to complete the same number of generations is shorter.

Even though CGPWNN requires less CPU time compared to CGPANN, it still produces better and equivalent accuracy results. The strength of CGPWNN is the wavelet functions and the CGP representation itself. Such wavelet function modifies input in a manner that has an equivalent effect of multiple neurons together.

Table 7 shows the performance of different classifiers in classifying the breast mass dataset. We can see that the proposed classifiers have outperformed most of the so-called *standard classifiers*, i.e. those not based on clustering or ensemble architectures.

6 Conclusions

This work compared existing (CGPANN, NEAT) and novel neuroevolutionary algorithms (CGPWNN) for their ability to classify malignant and benign patterns in digital mammograms.

CGPWNN achieved a classification accuracy of 89.57%, while CGPANN and NEAT reached 89.11%, respectively. Three features were consistently selected during the evolutionary process. *Patient age*, *mass margin* and *mass shape* thus play an important role in the correct classification of tumours. The CGPWNN algorithm was also found to be less computationally expensive and managed to search for higher quality solutions faster, compared to the other approaches.

The proposed technique was also found to perform comparatively to other techniques mentioned in literature and outperformed most of the standard classification algorithms.

Currently, a predefined structure for the search space is provided to the algorithm. In our future research, we intend to investigate a developmental form of CGPWNN that would evolve with each time-step. The performance of CGPWNN shall also be exploited further by introducing an adaptive threshold gene and the effect of having no bias. In addition, we plan to test the algorithm on other biomedical benchmark case studies.

7 Acknowledgments

We would like to acknowledge Dr. Ping Zhang, Research Scientist at CSIRO, for sharing the dataset features for research purposes.

References

Alexandridis, A. & Zapranis, A. (2013), 'Wavelet neural networks: A practical guide', *Neural Networks* **42**, 1–27.

Beyer, H. & Schwefel, H. (2002), 'Evolution strategies: A comprehensive introduction', *Natural Computing* **1(1)**, 3–52.

Billings, S. A. & Wei, H. (2005), 'A new class of wavelet networks for nonlinear system identification', *IEEE Transactions on Neural Networks* **16(4)**, 862–874.

Breast Cancer Care WA; accessed June (2014), `http://www.breastcancer.org.au/`.

Breiman, L., Friedman, J. H., Olshen, R. A. & Stone, C. J. (1984), *Classification and Regression Trees*, Pacific Grove:Wadsworth, Belmont, CA.

Byun, H. & Lee, S.-W. (2002), Applications of support vector machines for pattern recognition: A survey, *in* 'Proceedings of the First International Workshop on Pattern Recognition with Support Vector Machines', SVM '02, Springer-Verlag, London, UK, pp. 213–236.

Table 7: Comparison of different classifiers on the DDSM using 6 features

Algorithm	**Accuracy(%)**	**Sensitivity**	**Specificity**	**References**
		Standard classifiers		
LCA	87.00	80.50	93.90	(Pour et al. 2012)
AANN	91.00	90.00	92.00	(Panchal & Verma 2006)
SVM	87.50	88.40	91.60	(McLeod & Verma 2011)
K-means	84.50	–	–	(Verma et al. 2009*b*)
SOM	76.00	–	–	(Verma et al. 2009*b*)
NN	90.00	91.60	88.40	(Pour et al. 2012)
CART	91.00	–	–	(Kumar, Zhang, & Verma 2006)
C5.0	89.00	–	–	(Kumar, Zhang, & Verma 2006)
GANN	89.00	–	–	(Kumar, Zhang, & Verma 2006)
BPNN	88.00	–	–	(Kumar, Zhang, & Verma 2006)
		Ensemble & clustering classifiers		
SCBDL	97.50	97.50	97.50	(Verma et al. 2009*a*)
SCNN	94.00	97.83	90.74	(Verma et al. 2009*b*)
MCSVM	94.50	94.00	94.00	(McLeod & Verma 2011)
NN Ensemble	99.00	–	–	(McLeod & Verma 2013*a*)
LCA Ensemble	94.00	82.70	95.20	(Pour et al. 2012)
		Neuroevolutionary classifiers		
CGPANN	89.11	88.45	92.00	-
CGPWNN	89.57	84.90	95.64	-
CGPWNN-NL	89.57	85.38	94.38	-
NEAT	89.11	86.67	92.48	-

Cao, L., Hong, Y., Fang, H. & He, G. (1995), 'Predicting chaotic time series with wavelet networks', *Physica* **D85**, 225–238.

Chen, Y., Yang, B. & Dong, J. (2006), 'Time-series prediction using a local linear wavelet neural wavelet', *Neurocomputing* **69**, 449–465.

Digital Database for Screening Mammography; accessed May (2014), `http://marathon.csee.usf.edu/Mammography/Database.html`.

Fang, Y. & Chow, T. (2006), Wavelets based neural network for function approximation, *in* 'Advances in Neural Networks ISNN, Lecture Notes in Computer Science (LNCS)', Vol. 3971, Springer Berlin Heidelberg, pp. 80–85.

Hasiewicz, Z. (1997), Wavelet neural network for density estimation, *in* 'Proceedings of Third Conference on Neural Networks and Their Applications', pp. 136–141.

Heath, M., Bowyer, K., Kopans, D., Kegelmeyer, W., , Moore, R.and Chang, K. & MunishKumaran, S. (1998), Current status of the digital database for screening mammography, *in* 'Proceedings of the Fourth International Workshop on Digital Mammography', Kluwer Academic Publishers, pp. 457–460.

Heath, M., Bowyer, K., Kopans, D., Moore, R. & Kegelmeyer, W. (2001), The digital database for screening mammography, *in* 'Proceedings of the Fifth International Workshop on Digital Mammography', Medical Physics Publishing, pp. 212–218.

Kadambe, S. & Srinivasan, P. (2006), 'Adaptive wavelets for signal classification and compression', *International Journal of Electronics and Communications* **60**, 45–55.

Khan, M., Khan, G., Ahmad, A. & Miller, J. (2013), 'Fast learning neural networks using cartesian genetic programming', *Neurocomputing* **121**, 274–289.

Kumar, K., Zhang, P., & Verma, B. (2006), Application of decision trees for mass classification in mammography, *in* '2nd International Conference on Natural Computation , Advances in Natural Computation and Data Mining', Xidian University Press, China, pp. 365–375.

Kumar, K., Zhang, P. & Verma, B. (2006), Application of decision trees for mass classification in mammography, *in* 'Proceedings of Advances in Natural Computation and Data Mining', Xidian University Press, China, pp. 365–375.

McLeod, P. & Verma, B. (2010), A classifier with clustered sub classes for the classification of suspicious areas in digital mammograms, *in* 'International Joint Conference on Neural Networks (IJCNN)', IEEE, Barcelona, pp. 1–8.

McLeod, P. & Verma, B. (2011), Multi-cluster support vector machine classifier for the classification of suspicious areas in digital mammograms, *in* 'International Journal of Computational Intelligence and Applications', Vol. 10(4), Imperial College Press, pp. 481–494.

McLeod, P. & Verma, B. (2012*a*), Clustered ensemble neural network for breast mass classification in digital mammography, *in* 'World Congress on Computational Intelligence (WCCI)', IEEE, Brisbane Australia, pp. 1–6.

McLeod, P. & Verma, B. (2012*b*), A multilayered ensemble architecture for the classification of masses in digital mammograms, *in* 'AI 2012: Advances in Artificial Intelligence - 25th Australasian Joint Conference', Vol. 7691, Springer Berlin Heidelberg, Sydney, Australia, pp. 85–94.

McLeod, P. & Verma, B. (2013*a*), Effects of large constituent size in variable neural ensemble classifier for breast mass classification, *in* 'Neural Information Processing - 20th International Conference ICONIP', Vol. 8228, Springer Berlin Heidelberg, Daegu, Korea, pp. 525–532.

McLeod, P. & Verma, B. (2013*b*), Variable hidden neuron ensemble for mass classification in digital mammograms, *in* 'IEEE Computational Intelligence Magazine', Vol. 8(1), IEEE, pp. 68–76.

Miller, J. & Smith, S. (2006), 'Redundancy and computational efficiency in cartesian genetic programming', *IEEE Transactions on Evolutionary Computation* **10(2)**, 167–174.

Miller, J. & Thomson, P. (2000), Cartesian genetic programming, *in* 'European Conference on Genetic Programming, Lecture Notes in Computer Science (LNCS)', Vol. 1802, Springer-Verlag, pp. 121–132.

Muhlenbein, H. (1990), 'Limitations of multi-layer perceptron networks - steps towards genetic neural networks', *Parallel Computing* **14(3)**, 249–260.

Panchal, R. & Verma, B. (2006), 'Neural classification of mass abnormalities with different types of features in digital mammograms', *International Journal of Computational Intelligence and Applications* **6(1)**, 61–75.

Pour, S., McLeod, P., Verma, B. & Maeder, A. (2012), Comparing data mining with ensemble classification of breast cancer masses in digital mammograms, *in* 'Second Australian Workshop on Artificial Intelligence in Health: AIH 2012, held in conjunction with the 25th Australasian Joint Conference on Artificial Intelligence', The Netherlands, Tilburg University, Sydney, Australia.

Quinlan, J. (1993), *C4.5: Programs for Machine Learning*, Morgan Kaufmann, ISBN 1-55860-238-0.

RuleQuest-Research (2014), 'C5.0: An informal tutorial; accessed august', `http://www.rulequest.com/see5-unix.html`.

Stanley, K. & Miikkulainen, R. (2002), Efficient reinforcement learning through evolving neural network topologies, *in* 'GECCO', Vol. 9, San Francisco, Morgan Kaufmann, pp. 567–577.

Steinberg, D. & Colla, P. (1997), *CART - Classification and Regression Trees*, San Diego, CA:Salford Systems.

Steinberg, D. & Golovnya, M. (2006), *CART 6.0 User's Manual*, San Diego CA:Salford Systems.

Subasi, A., Alkan, A., Koklukaya, E. & Kiymik, M. K. (2005), 'Wavelet neural network classification of eeg signals by using ar model with mle pre-processing', *Neural Networks* **18**, 985–997.

Vassilev, V. & Miller, J. (2000), The advantages of landscape neutrality in digital circuit evolution, *in* 'International Conference on Evolvable Systems,Lecture Notes in Computer Science (LNCS)', Vol. 1801, Springer, pp. 252–263.

Verma, B. (2008), 'Novel network architecture and learning algorithm for the classification of mass abnormalities in digitized mammograms', *Artificial Intelligence in Medicine* **42(1)**, 67–79.

Verma, B., McLeod, P. & Klevansky, A. (2009*a*), 'Classification of benign and malignant patterns in digital mammograms for the diagnosis of breast cancer', *Expert Systems with Applications* **37(4)**, 3344–3351.

Verma, B., McLeod, P. & Klevansky, A. (2009*b*), 'A novel soft cluster neural network for the classification of suspicious areas in digital mammograms', *Pattern Recognition* **42(9)**, 1845–1852.

Yu, T. & Miller, J. (2001), Neutrality and the evolvability of boolean function landscape, *in* 'European Conference on Genetic Programming, Lecture Notes in Computer Science (LNCS)', Vol. 2038, Springer, pp. 204–217.

Yu, T. & Miller, J. (2002), Finding needles in haystacks is not hard with neutrality, *in* 'European Conference on Genetic Programming, Lecture Notes in Computer Science (LNCS)', Vol. 2278, Springer, pp. 13–25.

Zhang, P., Doust, J. & Kumar, K. (2010), Presenting a simplified assistant tool for breast cancer diagnosis in mammography to radiologists, *in* 'Medical Biometrics - Second International Conference ICMB', Vol. 6165, Springer, Hong Kong China, pp. 363–372.

Zhang, P. & Kumar, K. (2006), Analyzing feature significance from various systems for mass diagnosis, *in* 'International Conference on Computational Intelligence for Modelling Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMA-IAWTIC)', IEEE, pp. 141–146.

Zhang, P., Kumar, K. & Verma, B. (2005), A hybrid classifier for mass classification with different kinds of features in mammography, *in* 'Fuzzy Systems and Knowledge Discovery - Second International Conference, FSKD', Vol. 3614, Springer, Changsha, China, pp. 316–319.

Zhang, P., Verma, B. & Kumar, K. (2005), Neural vs. statistical classifier in conjunction with genetic algorithm based feature selection, *in* 'Pattern Recognition Letters', Vol. 26(7), pp. 909–919.

Zhang, Z. (2007), 'Learning algorithm of wavelet network based on sampling theory', *Neurocomputing* **71(1)**, 224–269.