Scientia Iranica, Vol. 14, No. 6, pp 625–630
© Sharif University of Technology, December 2007

SCIENTIA
IRANICA

Coevolution of Input Sensors and Recognition System to Design a Very Low Computation Isolated, Word Speech Recognition System

R. Halavati[1] **and S. Bagheri Shouraki***

Appropriate sensors are a crucial necessity for the success of recognition systems. Nature has always coevolved sensors and recognition systems and this can also be done in artificially intelligent systems. To get a very fast isolated word speech recognition system for a small embedded speech recognizer, an evolutionary approach has been used to create together the required sensors and appropriate recognition structures. The input sensors are designed and evolved through inspiration by the human auditory system and the classification is done by artificial neural networks. The resulting system is compared with a widely used speech recognition system, and the results are quite satisfactory.

INTRODUCTION

There is strong biological evidence that current human sensors have evolved and been optimized for our current way of life. Evolution has chosen a set of appropriate sensors for better dealing with the external world [1] and the evolution of sensors and processors has always been together, as either one of which without the other gives no evolutionary gain. Therefore, one can say that the ability to recognize the complex inputs of the external world, such as voice and vision, is not only due to computational power, but also, because of the correct set of sensors and preprocessors through which one absorbs the data.

Many recognition and classification approaches that are used in the current trends of artificial intelligence are inspired by natural approaches or insights into our conscious data processing. Examples of these approaches are: Neural networks [2-4], fuzzy systems [5-7], genetic and evolutionary algorithms [8,9], symbiotic based evolutionary algorithms [10,11] and artificial immune systems [12,13], as inspired directly by nature. There are also approaches, such as rule based systems, hidden Markov models [14-16] and case based reasoning [17,18], with based ideas taken from conscious data processing.

But, when it comes to feature selection, this is not the common approach and the features are usually computed using highly complex mathematics. It is clear that the computational powers of human beings and digital computers are quite different in nature and one may need different approaches for solving human-centric problems when they are faced by computers. However, a similar coevolution of sensors and a recognition approach can also be appropriate for artificial recognition systems.

In order to design and implement a very low computation speech recognition system for an embedded device, a co-evolutionary approach has been used, in which a set of appropriate sensors are formed to work together with some neural network classifiers. In this approach, the general structure of sensors and classifiers are inspired by the human auditory system, but the exact specification and structure is left to evolution.

In the rest of the paper, first the basic idea behind the sensors' structure will be presented. Then, the actual implementation details are included and the authors' genetic modeling and the evolutionary approach are presented. After that, the experimental

[1]. *Department of Computer Engineering, Sharif University of Technology, Tehran, I.R. Iran.*
*. *Corresponding Author, Department of Computer Engineering, Sharif University of Technology, Tehran, I.R. Iran.*

results and, finally, conclusions and future works are described.

THE RATIONALES BEHIND THE APPROACH

The human ear is composed of three sections. The external ear is in charge of gathering sound waves and reflecting then towards the middle ear. The middle ear is mainly in charge of some mechanical amplifications and medium transfer, so that the signal reaches the inner ear, where the processing begins. The main hearing organ responsible for sound perception is the cochlea. The cochlea is a spiral structure with two and a half turns and around one inch long if unfolded. The membrane of the cochlea is covered with hair cells, which detect sound activity and respond to it. The cochlea includes hair cells with selective frequency activity [19], so that each hair cell is activated mostly by a certain frequency and the activation level decreases as the frequency of the incoming sound gets further from this base frequency [20,21]. Also, it must be noted that, like all other neurons, hair cells have activity limitations, due to the required chemical charges at their synapses. Therefore, there is a minimum required rest time between two consecutive firings [22].

Based on these facts, a set of sensors are assumed, each responding to a certain range of frequencies, taking the average or maximum of the sensed frequencies, similar to the selective activation of hair cells. Also, as the hair cells have consecutive activation limits, the sensors can also sense and record the perceived signals in time windows and produce an output, based on an aggregation of perceived signals during this time. Once these first layer sensors have computed their outputs, the values can be given to a neural network to be classified.

The number of sensors, the position of each sensor, its range, its activation rules and the structure and features of the neural network can be subjects of a coevolution between sensors and the classification system.

THE IMPLEMENTATION DETAILS

Based on the proposed idea, the system consists of two subsystems. The first one is composed of the sensors, which absorb the speech signal and convert it into intermediate data and the second one is the neural network that learns and classifies the given data.

Sensors

The human ear uses a mechanical approach to separate different sound frequencies by using frequency dependent hair cells. But, in conventional speech recognition tools, there is just one recording device, which senses all frequencies together and, if one needs frequency amplitudes separately, this must be done by a computational approach. The most common approach to detect the amplitude of different frequencies is to use a short-term furrier transform. As the activity range of each sensor will be defined later, a furrier transform can be made with a linear scale and the exact positioning of the sensors can be left to later stages of the algorithm. Once the evolution is completed, the furrier transform can be done by the scale provided by the final positioning of the sensors.

Each input sensor has a definite frequency range from which it can absorb data. Sensors may have different bandwidths, may overlap with each other and may leave some frequency range unsampled.

The sensors are applied over time windows; therefore, the input of each sensor is all the frequency amplitudes that lie in the rectangle that is produced by its coverage range and time window. The output of each sensor is passed through a Max-N operator [23]. The operator takes the average of the maximum, N, percent of its inputs by sorting inputs in descending order and taking the average of the top, N, percent of them. Using different values of N, the operator can vary between average and maximum operator.

Figure 1 presents the spectrogram and output results of a set of sensors that have been designed, based on the MEL scale, as a possible start position of the sensors evolution.

Classification Network

To classify the sensors' outputs, a set of feed forward neural networks with an error back propagation training algorithm [24] was assumed, each for the acceptance of one word. Each network has an input layer with the size of its sensors multiplied by the number of time windows and an output layer with one node specifying whether the input belongs to that class or not. The number of intermediate layers, their sizes and their activation functions are left to evolution. The activation function can be either Sigmoid or Gaussian. Figure 2 presents a sample neural network.

General Structure

Putting the above components together, the complete system starts the recognition task by passing the input spectrogram through sensors and getting a matrix of the intermediate data. Then, this data is given to classifying neural networks and the class whose network output is the maximum, is chosen as the correct class. The complete schema is presented in Figure 3.

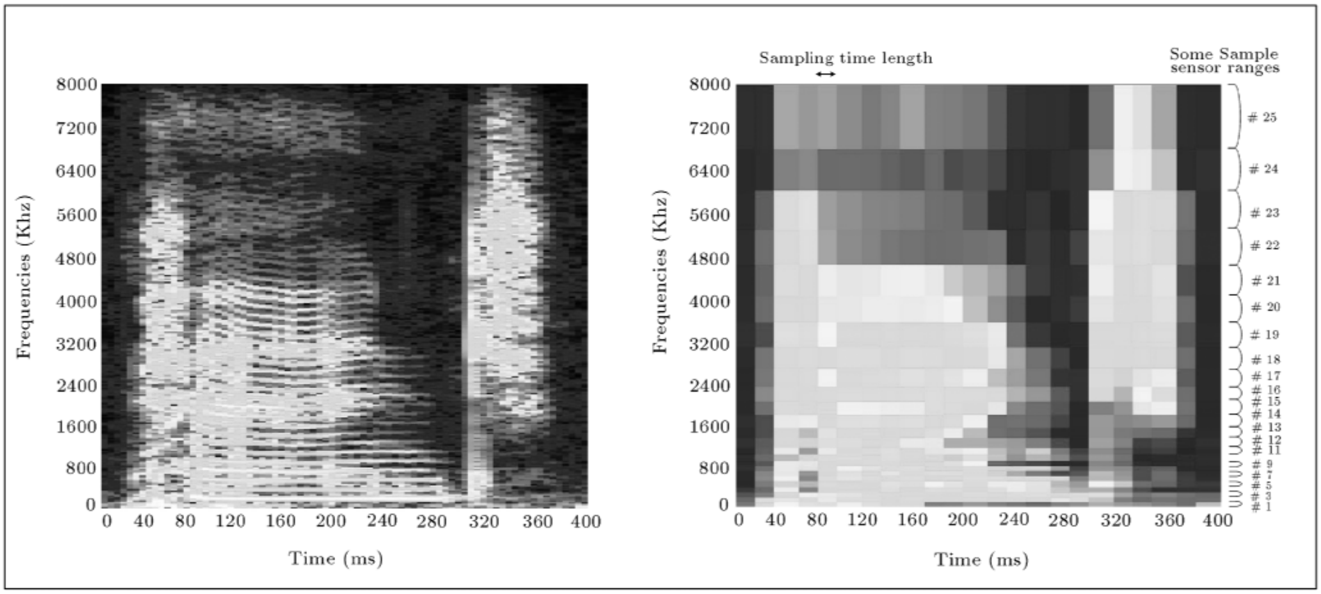

Figure 1. A sample spectrogram of speech signal (left) and sampled inputs using a set of sensors, designed based on the MEL scale (right) [25].

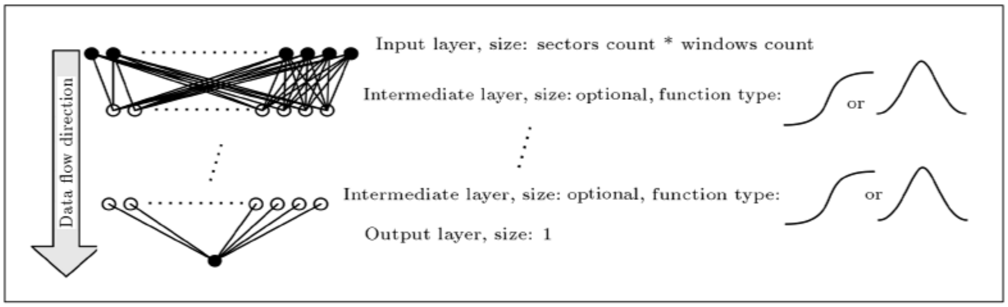
Data flow direction
Input layer, size: sectors count * windows count
Intermediate layer, size: optional, function type:
or
Intermediate layer, size: optional, function type:
or
Output layer, size: 1

Figure 2. The structure of the recognition neural networks.

GENETIC STRUCTURES AND EVOLUTIONARY TRAINING

To design the previously stated sensors and the classification networks, a coevolution between sensors and networks was used. To do so, the sensors and networks are coded in common chromosomes and a standard genetic algorithm is run to optimize these settings.

Chromosomes

The following genes compose one chromosome:

- The number of sensors,
- Time windows' count,
- For each sensor:
  - The first frequency in the range,
  - The last frequency in the range,
  - The parameter of the Max-N operator.
- The number of the classification networks' intermediate layers (all networks have the same number of intermediate layers, for the sake of simplicity),
- For each intermediate layer:

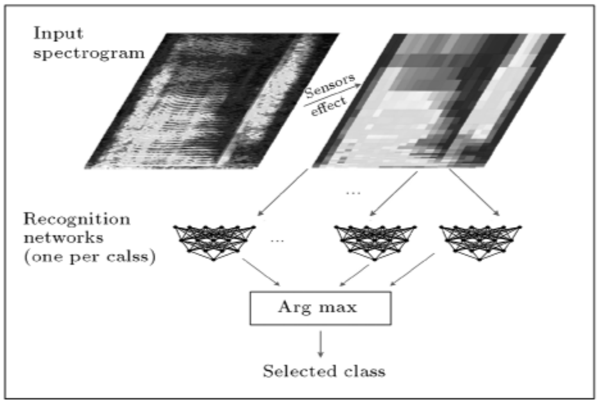

Figure 3. The general structure of a recognition system.

- Number of nodes in this layer,
- Activation function of cells, which can be either Sigmoid or Gaussian.

Fitness Function

To compute the fitness of each chromosome, its phenotype (the constructed sensors and networks) is trained using an error back propagation algorithm and the negative of the final recognition error rate is returned as the fitness value.

Mutation Operator

The following mutation operators were used during the creation of new offspring with equal probabilities:

- Alteration of one sensor's first or last frequency,
- Alteration of one sensor's operation type (Max-N's parameter),
- Duplication of one sensor,
- Merging of two sensors,
- Dropping of one sensor,
- Alteration of number of neural networks' intermediate layers,
- Alteration of one intermediate layer's number of cells,
- Alteration of one intermediate layer's activation function.

Cross-Over Operator

During a crossover, two chromosomes swap their sensors, keeping their own network structures.

EVALUATION AND EXPERIMENTAL RESULTS

The above structure was tested on two different test cases. The first set was FARS DAT (FARS DAT includes a variety of Farsi speech data uttered by 304 native speakers who differ from each other with regard to age, gender, dialect and educational level. Each speaker uttered twenty sentences in two sessions. The speech was collected in an acoustic booth of the Linguistics Laboratory of the University of Tehran); a standard multi-speaker Persian speech database, which is widely used in Persian speech recognition systems, and a random set of 250 words, spoken by 10 different speakers, half-male and half-female, were chosen. For the second test, 100 words were recorded by one 21-year-old male speaker and each word repeated 10 times. The words were selected from the literature book of the first year of primary school in Iran. Both sample sets were recorded by a 16 kHz sampling rate and spectrograms were computed by 512 frame windows and 384 frame window overlaps (128-frame step size).

To evolve the sensor/networks with genetic algorithms, a population of 100 chromosomes were used, 10% elitism, 60% crossover rate and 60% mutation rate. The initial design of the sensors was based on the MEL scale, but for diversity, the mutation operator was applied 100 times on each chromosome before starting the evolution. The initial structure of the neural network was randomly selected with 1-3 intermediate layers, each having 1-100 nodes. The parameter of the Max-N operator was set to 10%-40%, randomly.

To force the evolution to find cost-efficient designs, each chromosome had only 20 seconds for training its networks during fitness computations and, therefore, there was both one selection pressure for better recognition and another pressure for training speed, which could result in smaller sensors and network structure and, therefore, faster recognition speed. The 20 second limit was practically chosen.

In all runs, the training was with clean data, but testing was done with 20, 10, or 0 db SNRs of additive white or babble noise. The babble noise was recorded in a computer lab with about 30 people. 70% of each test set was used for training and 30% percent for testing. In the first test case, which had different speakers, the selection was speaker based (30% of the speakers were randomly chosen for the test).

To compare the recognition rate of the resulting system with standards of speech recognition, the same test and train cases were given to a speech recognition system which used MFCC features [18] and the HMM [18,26] recognition system.

Figures 4 and 5 present the comparison results of the recognition rate of the two systems and Figure 6 presents their recognition speeds.

As depicted in the diagrams, the evolved approach has a recognition rate almost the same as the MFCC-HMM approach, but much better results when the amount of noise, either babble or white, increases. Both systems have an almost 100% recognition result when testing is done with clean data, but with an increment in noise level, the recognition rate of both systems decreases. However, the decrement of the new hybrid approach is much slower than MFCC-HMM, as the new method reaches 55% to 75% correct recognition with the existence of 0 db noise, while the other has 5% to 39% correct results. Hence, it can be concluded that the new approach has been quite able to learn the patterns and ignore environmental noise.

The next test, depicted in Figure 6, presents the speed comparison of the two approaches. In this test, 10 minutes of clean samples, separated into 1000 files, was given to both systems to recognize. The MFCC feature extraction (not including spectrogram

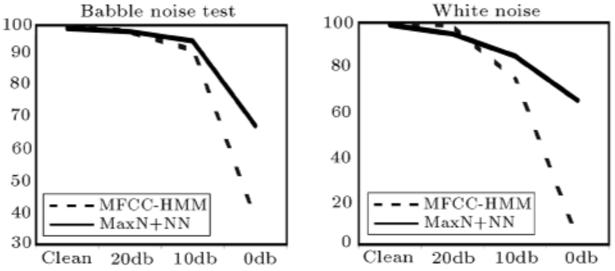

Figure 4. Single speaker recognition results for the presented approach vs. MFCC-HMM approach.

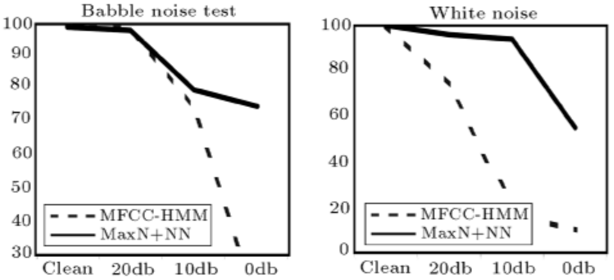

Figure 5. Multiple speaker recognition results for the presented approach vs. MFCC-HMM approach.

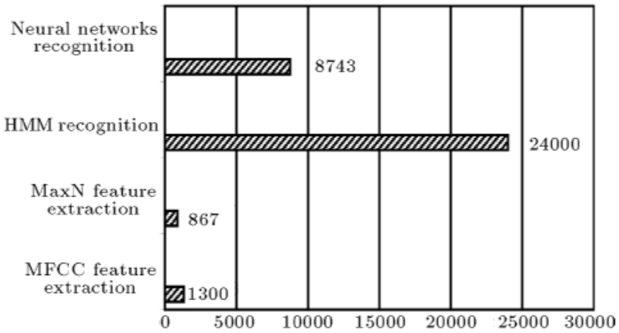

Figure 6. Processing time for classifying 10 minutes of recorded speech in milliseconds.

computation) took a little less than double the time of the Max-N used and the HMM took about three times more time than the neural networks.

Finally, to have a measure of the relation between recognition speed and recognition accuracy for selection of an appropriate system for the embedded hardware, a simple multi-objective optimization algorithm was used [27]. In this scenario, there were 100 separate nests in the gene pool. Once the fitness of one chromosome is computed, it moves to one of the nests, based on its recognition rate (the first nest accepts recognition rates in the [0..1) range, the second in the [1..2) range and the last in the [100..100] range). Then, to generate the next generation, the chromosomes in each nest are sorted, based on their recognition time, and the higher computation chromosomes are dropped. The new generation is created by mutations and crossovers over the contents of these nests.

As the result of this run, a set of chromosomes were obtained with different recognition speeds and accuracy. Figure 7 presents the results. As depicted in Figure 6, the chromosome with a recognition rate of 93 percent takes only 1 microsecond, while a 2 percent better accuracy requires triple time. Therefore, if the extra 2-3 percent accuracy were not crucial (and, usually, it is not, as the recognition is based on current context), one could have an appropriate system with 3 times more recognition speed.

CONCLUSIONS AND FUTURE WORKS

The coevolution of recognition systems and sensors is an approach, which has been tested and proved in nature and is now being used in this paper to design a very low computation speech recognizer algorithm for an embedded system. In this approach, the general structure of sensors is inspired by the human auditory system and the recognition system is a set of neural network classifiers.

The evolved system is tested on two speech databases and the results are compared with a widely used top-level speech recognition system, namely; MFCC features and an HMM recognizer. As stated in the previous section, the results show a system with a little less accuracy in comparison with top-level recognition systems, but using about 300 times less necessary computation, which was the bottleneck in the design of the hardware embedded system.

The approach has no language dependencies, as no model was used related to the Persian language. Implication of the sensors can be easily used in conventional speech recognition systems, but, the general approach is not suitable for continuous speech recognition, as it is heavily dependent on fixed time windows for each word.

In comparison with general speech recognition systems, a major simplification of this model are the fixed time windows. Using this simplification, one

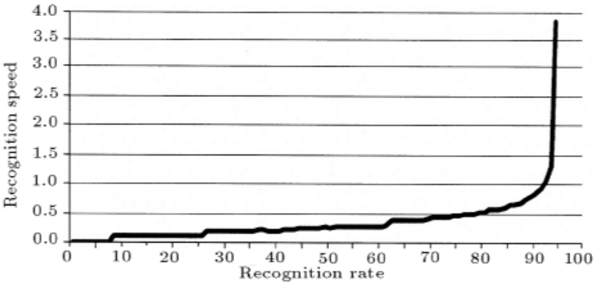

Figure 7. The relation between the recognition speed and the recognition accuracy of possible sensor-network structures. The horizontal axis represents recognition accuracy in percent and the vertical range represents recognition speed in microseconds on a Pentium IV 2.4 GHz.

can only cope with intonations almost similar to the recorded samples. The results on the two test sets were quite satisfactory and a broad set of intonations was not an important issue in the hardware device. But, as a next step in this approach, the authors are working on a more elastic recognition system to be used, instead of neural networks with fixed time windows.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Hossein Sameti and Mr. Bagher Baba-Ali for their help with test cases and comparison with the MFCC-HMM system and Ms Arpineh Choulakian, Ms Mina Rezagpour and Mr. Hossein Tajik for some parts of the implementation and testing.

REFERENCES

1. Rose, S.P, *The Conscious Brain, Random House Inc Publication*, Chapters 5 and 6 (1973).
2. Hebb, D.O., *The Organization of Behavior*, John Wiley & Sons, New York, USA (1949).
3. Hopfield, J.J. "Neural networks and physical systems with emergent collective computational abilities", in *Proc. Nat'l Academy of Sciences*, USA, **79**, pp 2.554-2.558 (1982).
4. Kohonen, Y., *Self Organization and Associative Memory*, 3rd Ed., Spinger-Verlag, New York, USA (1989).
5. Zadeh, L.A. "Toward a theory of fuzzy information granulation and its centricity in human reasoning and fuzzy logic", *Fuzzy Sets and Systems*, **90**(2), pp 111-127 (1997).
6. Mamdani, E.H. and Assilian, S. "An experiment in linguistic synthesis with a fuzzy logic controller", *International Journal of Man-Machine Studies*, **7**(1), pp 1-13 (1974).
7. Kandel, A., *Fuzzy Expert Systems*, CRC Press (1992).
8. Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley (1989).
9. Venturini, G. "SIA: A supervised inductive algorithm with genetic search for learning attribute based concepts", in *Proc. European Conference on Machine Learning*, pp 280-296, Vienna, Austria (1993).
10. Watson, R.A. and Pollack, J.B. "Incremental commitment in genetic algorithms", *Proceedings of GECCO 1999*, Banzhaf, et al., Eds., Morgan Kaufmann, pp 710-717 (1999).
11. Watson, R.A. and Pollack, J.B. "Symbiotic combination as an alternative to sexual recombination in genetic algorithms", *Proceedings of Parallel Problem Solving from Nature (PPSN VI)*, pp 425-434 (2000).
12. Ada, G.L. and Nossal, G. "The clonal selection theory", *Scientific American*, **257**(2), pp 50-57 (1987).
13. Dasgupta, D., *Artificial Immune Systems and Their Applications*, Ed., Springer-Verlag (1999).
14. Baum, L.E. and Petrie, T. "Statistical inference for probabilistic functions of finite state Markov chains", *Ann. Math. Stat.*, **37**, pp 1554-1563 (1966).
15. Baum, L.E. and Egon, J.A. "An inequality with applications to statistical estimation for probabilistic function of a Markov process and to a model for ecology", *Bull. Amer. Meteorol. Soc.*, **73**, pp 360-363 (1967).
16. Rabiner, L.R., Levinson, S.E. and Sondhi, M.M. "On the application of vector quantization and hidden Markov models to speaker-independent isolated word recognition", *Bell Systems Technical Journal*, **62**(4), pp 1075-1105 (Apr. 1983).
17. Yang, S. and Robertson, D. "A case-based reasoning system for regulatory information", in *Proc. IEE Colloquium on Case-Based Reasoning: Prospects for Applications*, Digest No: 1994/057, pp 3/1-3/3 (1994).
18. Rissland, E.L. and Skala, D.B. "Combining case-based and rule-based reasoning: A heuristic approach", in *Eleventh International Joint Conference on Artificial Intelligence*, IJCAI-89, pp 524-30, Detroit, Michigan (1989).
19. Goldstein, E.B., *Sensation and Perception*, Wadsworth Publishing Company, Belmont, California, pp 438-442 (1989).
20. Cosi, P. "Auditory modeling and neural networks", in *A Course on Speech Processing, Recognition, and Artificial Neural Networks*, Springer Verlag, Lecture Notes in Computer Science (1998).
21. Ghitza, O. "Auditory nerve representation as a basis for speech processing", in *Advances in Speech Signal Processing*, S. Furui and M. Sondhi, Eds., Marcel Dekker, New York, pp 453-485 (1991).
22. Brownell, W.E. "How the ear works - natures solutions for listening", *Volta Review*, **99**(5), pp 9-28 (1997).
23. Halavati, R., Shouraki, S.N. and Zadeh, S.H. "Recognition of human speech phones using a novel fuzzy approach", *Applied Soft Computation*, **7**, pp 828-839 (2007)
24. Rumelhart, D.E. and McClelland, J.L., *Parallel Distributed Processing: Exploring in the Microstructure of Cognition*, MIT Press, Cambridge, MA, USA (1986).
25. Stevens, S.S. and Volksman, J. "The relation of pitch to frequency", *American Journal of Psychology*, **53**, p 329 (1940).
26. Rabiner, L. and Juang, B.H., *Fundamentals of Speech Recognition*, Prentice Hall, Enlgewood Cliffs, New Jersey (1993).
27. Abbass, H.A. "Pareto neuro-evolution: Constructing ensemble of neural networks using multi-objective optimization", in *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, **3**, pp 2074-2080, IEEE Press, Canberra, Australia (December 2003).