Design of Near-Optimal Classifier Using Multi-Layer Perceptron Neural Networks for Intelligent Sensors

Nadir N. Charniya, *Member, IACSIT*

***Abstract*—The Multi-layer Perceptron Neural Networks (MLP NN) are well known for their simplicity, ease of training for small-scale problems, and suitability for online implementation. This paper presents the methodology and challenges in the design of near-optimal MLP NN based classifier with maximize classification accuracy under the constraints of minimum network dimension for implementation intelligent sensors.**

***Index Terms*—Classifier, neural networks, multi-layer percptron, intelligent sensors.**

I. Introduction

Neural networks have special features, such as, capability to learn from examples, adaptations, parallelism, robustness to noise, and fault tolerance. They learn how to separate classes of signals by examples used for training and, therefore, the user need not know exhaustively the properties of the signals. Their main advantages are the ability to generalize results of sensors obtained from known situations to unforeseen situations. Neural networks are intrinsically capable of recognizing a complex pattern [1].

Tian [2] in his work related to intelligent sensors mentioned that intelligent sensors are an extension of traditional sensors to those with advanced learning and adaptation capabilities. An intelligent sensor can incorporate features that enable it to compensate for systematic errors, system drift, and random errors produced due to system parameters or the characteristics of the sensor. There are several references related to applications of neural networks in intelligent sensors used to incorporate the functionalities, such as, classification, regression, linearization, error correction, noise compensation, prediction, etc. to the sensors making them intelligent. Kato and Mukai [3] developed an intelligent gas sensor system for discrimination and quantification of gases by a single semiconductor gas sensor in real-time. Charniya and Dudul [4] developed neural networks-based intelligent sensor system for the classification of material type even with the variation in the sensor parameter. For classification, near-optimal classifier models were designed to maximize accuracy under the constraints of minimum network dimension. One of the most powerful uses of neural networks is in function approximation (curve fitting). In this context, the usage of neural network techniques provides lower interpolation errors when compared with classical method of polynomial interpolation [5].

Neural networks can exactly identify a nonlinear system model from the inputs and outputs of a complex system, and do not need to know the exact relationship between inputs and outputs. A mathematical model of thermocouple has been established based on neural networks by [6]. Arpaia et al. [1] have proposed a low cost methodology using neural network based inverse model and the related application procedure for compensating the systematic error in nonlinear sensors affected by combined interfering parameters. A prototype of an eddy-current transducer for displacement measurements has been designed by implementing the proposed approach. The neural networks capability of correcting the nonlinear influence of the target material and area variations was experimentally verified.

Temperature drift errors are problems that affect the accuracy of measurement systems. When small amplitude signals from transducer are considered and environmental conditions of conditioning circuits exhibit a large temperature range, the temperature drift errors have a real impact in system accuracy. Neural networks based solution to overcome the problem of temperature drift errors of signal conditioning circuits has been proposed by [5]. Patra and Bos [7], and Pramanik et al. [8] have proposed a novel computationally efficient neural network for modeling of a pressure sensor operated in a dynamic environment. It has been found that neural network is capable of estimation of pressure quite accurately irrespective of nonlinear characteristics of the pressure sensor, and its temperature dependence.

Borecki [9] presented the construction and working principles of neural network based intelligent fiber-optic intensity sensor used for examining the concentration of a mixture in conjunction with water using neural networks. Application of the sensor has been proposed in wastewater treatment plant for selection of a treatment process. Neural networks are powerful signal processing tools in the area of numerical linearization of sensor characteristics with compensation of errors due to influence quantities. Postolache et al. [10] have reported the development of an intelligent turbidity and temperature-sensing unit for water quality assessment using neural networks. The proposed intelligent turbidity sensor using neural network permits to reduce the effect of detector-to-detector variability, light source intensity variations, and water absorbance.

II. Neural Networks As Classifier

The central problem in classification is to define the shape

Manuscript received December 16, 2012; revised January 29, 2013.

Nadir. N. Charniya is with the VES Institute of Technolgy, Collectors Colony, RC Marg, Chembur, Mumbai, pin 400074, India (e-mail: nncharniya@gmail.com).

DOI: 10.7763/IJMO.2013.V3.234

and placement of the boundary (decision surface) in the pattern space so that the class-assignment errors are minimized. The neural network builds discriminant functions from its Processing Elements (PEs) or neurons. Discriminant functions intersect in the input space defining a decision surface. The discriminant function evaluates every position in pattern space and produces larger value for one class and low values for all others. The neural networks topology determines the number and shape of the discriminant functions. The shapes of the discriminant functions change with the topology. One of the major advantages of neural networks is that they are sufficiently powerful to create arbitrary discriminant functions; therefore, neural networks can achieve optimal classification. The placement of the discriminant functions is controlled by the network weights. The weights are adjusted directly from the training data without any assumptions about the data's statistical distribution. Hence, one of the important issues in neural network-based classifier design is to utilize systematic procedures (a training algorithm) to modify the weights so that as accurate a classification as possible is achieved.

Neural classifiers have the advantage of reducing misclassifications among the neighborhood classes compared to linear classifiers (such as, Classification And Regression Trees (CART)). The CARTs are statistical structures that were proposed by [11]. The Fig. 1 shows a typical example of decision boundaries formed by linear classifiers and neural network-based classifiers for three distinct classes. The thick lines in the figure show the decision boundary obtained by using linear classifier, whereas, the thin curves show the decision boundary corresponding to the neural network-based classifier. It is depicted that the linear classifier gives relatively poor separation of the classes compared to the neural network-based classifier.

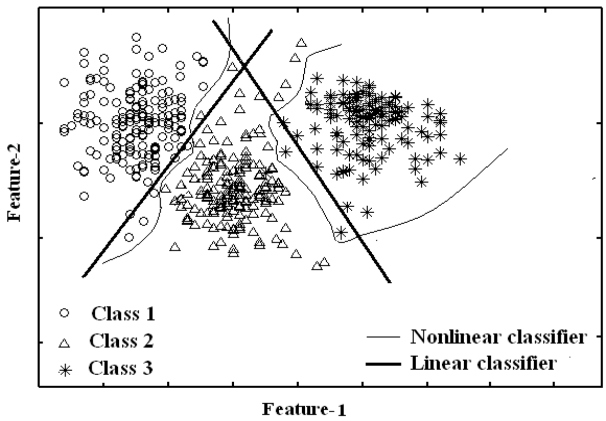

Fig. 1. Class separation for neural networks-based classifier and linear classifier

III. Design of MLP NN Based Classifier

The Multi-layer Perceptron Neural Network (MLP NN) is selected in many applications due its simplicity, suitability for online implementation, and ease of training for small-scale problems. The capability of MLP NN trained by error back-propagation algorithm (supervised training) has been successfully demonstrated in many applications for nonlinear function approximation, and classification with any degree of accuracy.

The near optimal classifiers using MLP NN should be meticulously designed using all generalization parameters. The generalization performance of the networks are validated meticulously on the basis of important parameters [12], [13] such as, mean square error (MSE), and percent classification accuracy (PCLA) on the testing instances, while *attempting different input data partitions*. Input data used in this paper belong to neural networks based intelligent sensor developed by the author [4]. A rigorous computer simulation has been carried out to design the near-optimal parameters of the MLP NN based classifiers. The classifier model is designed to maximize accuracy under the constraints of minimum network dimension.

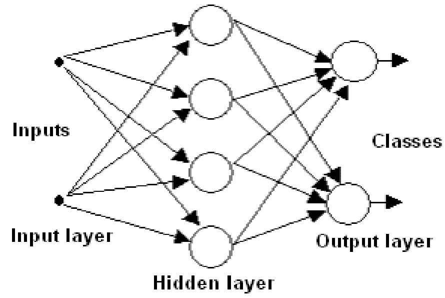

Fig. 2. Multilayer perceptron neural networks

Fig. 2 illustrates a typical MLP NN. The circles are the PEs arranged in layers. The left column is the input layer, the middle column is the hidden layer, and the right column is the output layer. The lines represent weighted connections (i.e., a scaling factor) between PEs.

The architecture of network is almost completely determined by problem specifications, including the specific number of inputs and outputs and the particular output signal characteristic. Number of network inputs is equal to number of problem inputs. Whereas, the number of neurons in the output layer is equal to problem outputs.

When designing a neural network, one should concentrate with the aspects such as, network topology, number of layers in the network, number of neurons or nodes per layer, choice of activation function in the neurons, learning algorithm to be adopted, numbers of iterations per pattern during training, network performance, final values of weights and biases, etc. The design of a near optimal MLP NN requires the determination of the activation functions and the thresholds of the PEs, as well as of the connection weights. The activation functions and the thresholds are defined by a recursive optimization procedure. The connection weights are computed by means of a learning algorithm.

The MLP NN based classifier models aims at maximum classification accuracy under the constraints of minimum network dimension. For this, the classifier has to be tested with several *different numbers of hidden units* and incremental results are obtained corresponding to how well the *different variants of back-propagation algorithm.*

Fig. 3 shows the training curves for a typical MLP NN trained with different variants of back-propagation training algorithms, such as, Step learning, (STEP), Momentum learning, Delta-Bar-Delta learning (DBD), Conjugate Gradient learning (CG), Levenberg Marquadt learning (LMQ), and Quick Propagation learning (QP) performed

[13]. Choose the training algorithm that converges faster with minimum oscillations during convergence. MOM depicts faster convergence with reasonable stability for a typical MLP NN.

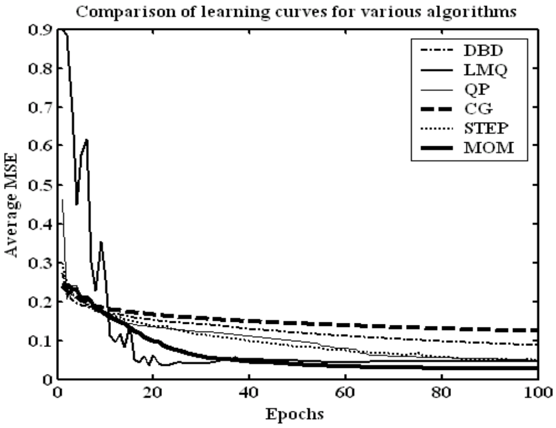

Fig. 3. Comparison of different learning curves for the training of MLP NN

During training of the neural networks, the learning curves for the training and validation data sets are plotted (Fig. 4). When the error on the validation set is lowest the network is deemed to have reached the generalization [12]. The trained networks are then challenged by test data sets whose classes are unknown. The generalization performance of the networks should be validated meticulously on the basis of important parameters such as, MSE and PCLA on the testing instances, while attempting different data partitions to improve the speed of learning and generalization ability, and avoidance of local minima.

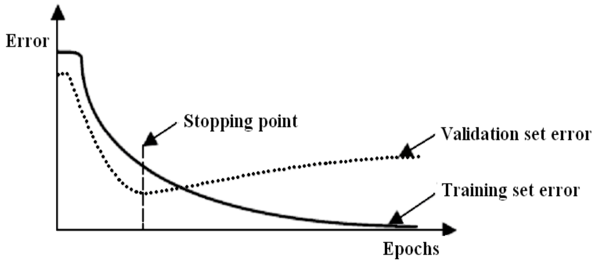

Fig. 4. Stopping point from learning curve for training and validation sets

Another issue of training the neural networks is how to choose the *initial weights*. The search must start some place on the performance surface (the total error surface plotted in the space of the system coefficients (weights)). That place is given by the initial condition of the weights. However, the network's PEs have saturating nonlinearities, so if the weight values are very large, the PE can saturate. If the PE saturates, the error that goes through becomes zero, and previous layers may not adapt. Small random weight values will put every PE in the linear region of the sigmoid at the beginning of learning.

Before training, the *entire data set is usually randomized* first. The initial conditions and other training parameters have a prominent effect on the learning performance of the network. In order to gauge the real performance of neural networks, it should be *re-trained a number of times with different random initialization of connection weights* [13]. This ensures true learning, helps *avoid local minima, and entails generalization*. The possible parameter variations chosen for the design of MLP NN are depicted in the Table I.

TABLE I: VARIABLE PARAMETERS OF NEURAL NETWORKS

Parameters	Typical Range
Number of hidden layers	(1 to 3)
Number of hidden neurons	(2 to 100)
Learning-rate parameter	(0 to 1)
Momentum constant	(0 to 1)
Transfer function of neurons in the network layers	tanh, sigmoid, linear tanh, linear sigmoid, linear
Learning rule	STEP, MOM, DBD, CG, LMQ, QP

Variation of average of PCLA with the number of hidden neurons, for the training, cross validation, and testing data set is plotted in a typical example shown in Fig. 5. Similarly, learning rate parameter, and momentum constant are determined [13]. The choice of these values was made as per the exhaustive experimentation for the training of the MLP NN for different values of these parameters. The values which maximum average PCLA should be chosen.

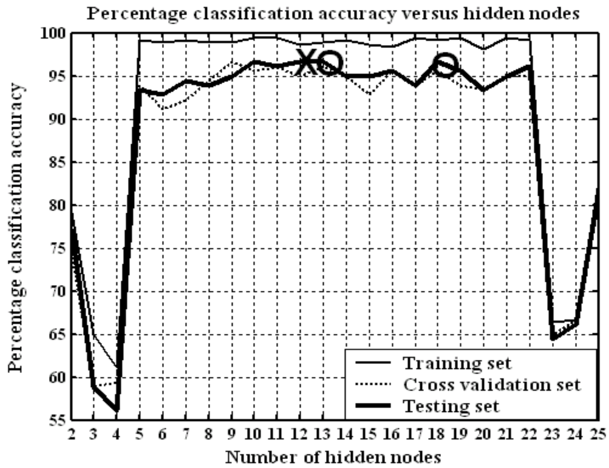

Fig. 5. Accuracy with the number of hidden neurons of a typical MLP NN for training, cross validation, and testing data sets

Further, the network should be trained with different transfer function in hidden layer and output layer. Fig. 6 shows example of a typical MLP NN that gave maximum classification accuracy for *tanh* transfer in hidden layer neurons.

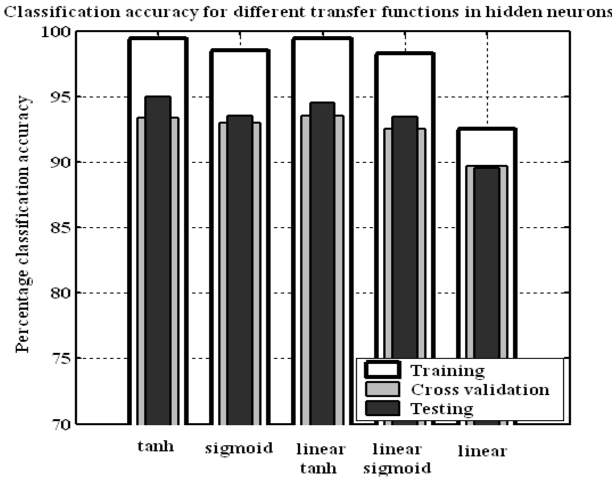

Fig. 6. Comparison of percentage classification accuracy of MLP NN for different neuron transfer functions in hidden layer.

Similar experimentation is carried out for different number of hidden layers with variations in number of neurons in them. Fig. 7 shows a plot of number of hidden layers versus the classification accuracy. It shows maximum accuracy for two hidden layer MLP NN. Increase in the number of hidden layer may not improve the performance of the classifier significantly. On the contrary, it takes more time for training because of higher computational complexity of the network, and the network looses generalization ability. In case the accuracy of MLP NN comes out to be same at different number of layers then test the robustness by adding equal percentage of Gaussian noise to the input data in order to select the optimal architecture.

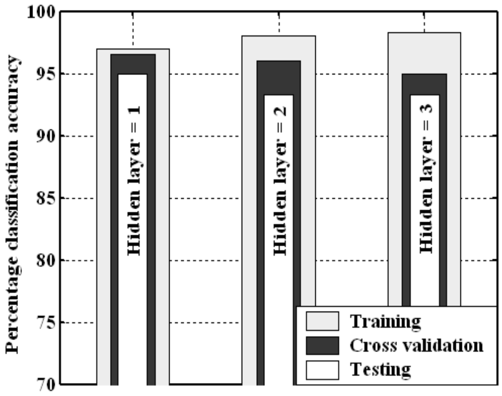

Fig. 7. Classification performance of the MLP NN on different number of hidden layers

IV. Hardware Implementation of Neural Networks

Implementation of neural networks can be accomplished using either analog or digital hardware. The digital implementation is more popular as it has the advantage of higher accuracy, better repeatability, lower noise sensitivity, better testability, and higher flexibility and compatibility with other types of preprocessors. On the other hand, analog systems are more difficult to be designed and can only be feasible for large-scale productions, or for very specific applications. The digital neural networks hardware implementations are further classified as: 1) Field-Programmable Gate Array (FPGA) based implementations; 2) Digital Signal Processor (DSP) based implementations; and 3) Application Specific Integrated Chip (ASIC) based implementations. DSP-based implementation is sequential and hence does not preserve the parallel architecture of the neurons in a layer. ASIC implementations do not offer re-configurability by the user, and are prohibitively costly in development. The advancement of FPGAs in recent years, allowing millions of gates on a single chip and accompanying with high-level design tools has allowed the implementation of very complex neural networks [14]. It allows the fast design of complex systems with the highest performance/cost ratio [15]. FPGA is the most suitable hardware for neural networks implementation as it preserves the parallel architecture of the neurons and can be flexibly reconfigured by the user as the application demands. It is also capable of supporting the dynamic creation and design modification of neural network topologies. It has performance speeds, fabrication area and precision closer to ASICs [16]. This type of realization makes the network stand alone and operate on a real-time fashion. Neural networks have been successfully implemented in digital hardware, such as FPGA, DSP, and microcontroller [17], [18].

Hardware implementation of MLP NN using FPGA has been described by [17]. Despite improvements in FPGA densities, the numerous multipliers in neural networks limit the size of the network that can be implemented using a single FPGA, thus making neural networks applications not viable commercially. The proposed implementation aimed at reducing resource requirement, without much compromise on the speed for online applications, so that larger neural networks can be realized on a single chip at a lower cost. The sequential processing of the layers in neural networks has been exploited in this paper to implement large neural networks using a method of layer multiplexing. Instead of realizing a complete network, only the single largest layer is implemented. The same layer behaves as different layers with the help of a control block. The control block ensures proper functioning by assigning the appropriate inputs, weights, biases, and excitation function of the layer that is currently being computed. MLP NNs have been implemented using Xilinx FPGA.

Yang and Paindavoine [18] describe three hardware implementations of RBF neural network-based real-time face tracking and identity verification model on embedded systems based, respectively, on FPGA, zero instruction set computer chips, and digital signal processor (DSP) TMS320C62 from Texas Instruments. Results of FPGA implementation have been presented in terms of hardware resources used and processing speed.

Depari et al. [19] have proposed an approach to estimate biaxial position with a pyro-electric sensor array. Pyro-electric sensors convert a flux of incident radiant energy into an electric signal, and an array of it was used for contactless displacement measurement by means of a light-spot cursor. A full DSP-based electronics approach to process signals from a pyro-electric sensor array has been proposed and experimentally evaluated to improve performances and simplify the calibration procedure if compared to an analog electronic circuit. The DSP TMS320C6711 based electronics was used to handle the excitation, acquisition, and processing that resulted in a compact solution. By means of coherent sampling and fast Fourier techniques, the light intensity of each array element can be measured. A DSP-based hardware has been also developed for different neural networks.

Not only the neural networks but also the required pre-processing and feature extraction techniques can be carried out by a precise and optimal digital hardware implementation for further compactness, portability, and reliability.

V. Conclusion

An exhaustive computer simulation should be carried out to design the optimal parameters of the neural networks. For ensuring the generalization and true learning performance, the classifiers should be trained and tested on different input data partitions. This is essential to remove any biasing and prejudice from the learning machine. Refinements are made to the resources used by the neural networks, with a result of a reduction in the training time and optimal architecture of the networks. Optimal trade-off between neural network accuracy and complexity is determined. Using this methodology, MLP NN network can be designed to maximize accuracy under the constraints of minimum network dimension so that its hardware implementation for intelligent sensors further requires minimum number of components to satisfy real time constraints and low power consumption.

REFERENCES

[1] P. Arpaia, P. Daponte, D. Grimaldi, and L. Michaeli, "ANN-based error reduction for experimentally modeled sensors," *IEEE Trans. on Instrumentation and Measurement*, vol. 51, no.1, pp. 23-30, Feb. 2002.

[2] G. Y. Tian, "Design and implementation of distributed measurement systems using field-bus based intelligent sensors," *IEEE Trans. on Instrumentation and Measurement*, vol. 50, no. 5, pp. 1197-1202, 2001.

[3] Y. Kato and T. Mukai , "A real-time intelligent gas sensor system using a nonlinear dynamic response," *Sensors and Actuators-B*, vol. 120, pp. 514-520, 2007.

[4] N. N. Charniya and S. V. Dudul, "Sensor for classification of material type and its surface properties using radial basis networks," *IEEE Sensors Journal*, vol. 8, no. 12, pp. 1981-1991, 2008.

[5] J. M. D. Pereira, O. Postolache, P. S. Girao, and M. Cretu, "Minimizing temperature drift error of conditioning circuits using artificial neural networks," *IEEE Trans. on Instrumentation and Measurement*, vol. 49, no. 5, pp. 1122-1126, 2000.

[6] Z. Y. Song, C. Y. Liu, and X. L. Song, "Mathematical model of thermocouple characteristic based on neural network," in *Proceedings of the 2nd International Conf. on Machine Learning and Cybernetics*, Xi'an, 2003, pp. 1404-1407.

[7] J. C. Patra and A. V. D. Bos, "Modeling of an intelligent pressure sensor using functional link artificial neural networks," *ISA Trans.*, vol. 39, pp. 15-27, 2000.

[8] C. Pramanik, T. Islam, and H. Saha, "Temperature compensation of piezoresistive micro-machined porous silicon pressure sensor by ANN," *Microelectronics Reliability*, vol. 46, pp. 343–351, 2006.

[9] M. Borecki, "Intelligent fiber optic sensor for estimating the concentration of a mixture - Design and working principle," *Sensors*, vol. 7, pp. 384-399, 2007.

[10] O. Postolache , J. M. D. Pereira , P. S. Girao, and H. Ramos , "Smart flexible turbidity sensing based on embedded neural network," in *Proc. 5th IEEE Conf. on Sensors*, Daegu, Korea, 2006, pp. 658-661.

[11] L. Breiman, J. Friedman, R. Olshen, and P. Stone, *Classification and regression trees*, Wadsworth, Belmont, CA, 1984.

[12] S. Haykin, *Neural network: A comprehensive foundation*, Pearson Prentice Hall, New Delhi, 2007.

[13] J. Principe, N. Euliano, and C. Lefebvre, *Neural and adaptive systems: Fundamentals through simulations*, Wiley, New York, 1999.

[14] H. S. Ng and K. P. Lam , "Analog and digital FPGA implementation of BRIN for optimization problems," *IEEE Trans. on Neural Networks*, vol. 14, no. 5, pp. 1413-1425, 2003.

[15] L. M. Reyneri, "Implementation issues of neuro-fuzzy hardware: Going towards hardware/software co-design," *IEEE Trans. on Neural Networks*, vol. 14, no. 1, pp. 176-194, 2003.

[16] H. Hikawa, "A digital hardware pulse-mode neuron with piecewise linear activation function," *IEEE Trans. on Neural Networks*, vol. 14, no. 5, pp. 1028-1037, 2003.

[17] S. Himavathi, D. Anitha, and A. Muthuramalingam, "Feedforward neural network implementation in FPGA using layer multiplexing for effective resource utilization," *IEEE Trans. on Neural Networks*, vol. 18, no. 3, pp. 880-888, 2007.

[18] F. Yang, and M. Paindavoine, "Implementation of an RBF neural network on embedded systems: real-time face tracking and identity verification," *IEEE Trans. on Neural Networks*, vol. 14, no. 5, pp. 1162-1175, 2003.

[19] A. Depari, P. Ferrari, V. Ferrari, A. Flammini , A. Ghisla, D. Marioli, and A. Taroni, "Digital signal processing for biaxial position measurement with a pyroelectric sensor array," *IEEE Trans. on Instrumentation and Measurement*, vol. 55, no. 2, pp. 501-506, 2006.

Nadir N. Charniya was born on September 30, 1966. He received the Master of Engineering degree in electronics from Victoria Jubilee Technical Institute, Mumbai, India, in 1995 and the Ph.D. degree in design of intelligent sensors using a neural network approach from Sant GadgeBaba Amravati University, Amravati, India, in 2010. He has about 22 years of teaching experience. He is currently working as a Professor, Department of Electronics and Telecommunications Engineering, VES Institute of Technology, Chembur, Mumbai, India. He has papers published in refereed international journals and international conference proceedings. He has chaired various conferences and delivered expert talk on signal processing techniques and their applications at various engineering colleges. His areas of interest include intelligent sensors and systems, neuro-computing techniques, signal processing, and their applications.

Dr. Charniya is a member of the Institute of Electronics and Telecommunication Engineers, the Indian Society for Technology in Education, India, International Association of Computer Science and Information Technology. He is recipient of various research and laboratory development grants from AICTE, New Delhi and Universities. He has been invited to work as a reviewer for papers submitted to the IEEE transactions and Elsevier international journals in relation to his field of interest. A project on robotics guided by him received the First Prize and the "Maharashtra State Engineering Design Award" from the ISTE, India. His Ph.D. work on intelligent sensors received the First Prize under Post P.G. category in the Maharashtra State Interuniversity Research Project Exhibition "AVISHKAR—2008."