33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

Optimization of Head Movement Recognition Using Augmented Radial Basis Function Neural Network

Mitchell Yuwono, A.M. Ardi Handojoseno, H.T. Nguyen, *Senior Member, IEEE*

Abstract—**For people with severe spine injury, head movement recognition control has been proven to be one of the most convenient and intuitive ways to control a power wheelchair. While substantial research has been done in this area, the challenge to improve system reliability and accuracy remains due to the diversity in movement tendencies and the presence of movement artifacts. We propose a Neural-Network Configuration which we call Augmented Radial Basis Function Neural-Network (ARBF-NN). This network is constructed as a Radial Basis Function Neural-Network (RBF-NN) with a Multilayer Perceptron (MLP) augmentation layer to negate optimization limitation posed by linear classifiers in conventional RBF-NN. The RBF centroid is optimized through Regrouping Particle Swarm Optimization (RegPSO) seeded with K-Means. The trial results of ARBF-NN on Head-movement show a significant improvement on recognition accuracy up to 98.1% in sensitivity.**

Keyword— **Unsupervised Clustering, Supervised Learning, Pattern Recognition, Head-movement, Neural-Network, Radial Basis Function, Regrouping Particle Swarm Optimization.**

I. Introduction

Head movement recognition strategies with neural network have been proven to be a comfortable, reliable and natural way of controlling power wheelchairs for people with severe mobility disabilities such as quadriplegia, cerebral palsy, etc. [1]-[3]. Accuracy and robustness are the main issues to be handled before it can be applied for real use. Head movement patterns essentially differ in many factors including speed, trajectory, noises, magnitude, hold time, and other tendencies depending on one's habit and level of handicap which significantly affects the reliability of the detector. Special care needs to be considered regarding pattern diversity and noise resistance.

A head-movement based wheelchair controller using MLP configuration by Nguyen & Craig showed a promising result of up to 95.8% of sensitivity [3]. Inspired by this research, we propose an optimization method using a hybrid RBF and MLP configuration which we call the Augmented Radial Basis Neural Network (ARBF-NN). This technique optimizes the efficiency of the RBF class separation using an MLP augmentation layer [4]. The RBF centroids and radii are optimized using K-means seeded RegPSO algorithm. Similar hybrid configurations of RBF/MLP for different applications have been explored in other researches [5], [6].

This paper has several objectives which are to enhance the reliability of head-movement recognition and to explore the capability of ARBF-NN for this particular problem.

This paper is organized as follows: section II presents an overview of the system, the unsupervised data clustering method is discussed in section III, the ARBF-NN is discussed in section IV, the Experimental results and analysis is highlighted in section V, and finally section VI provides a conclusion and future directions for the current project and proposed technique.

II. System Overview

The overall system can be seen in Fig.1. Input data for the system are sampled 3-D acceleration signals. The accelerometer module used in this project is RD-3152 MMA7260Q – Zstar2 from Freescale Semiconductor. It provides 3-axis acceleration readings using an MMA7260Q accelerometer set to ±1.5g sensitivity range. The sensor uses wireless communication based on ZigBee protocol 2.4GHz band to communicate to the receiver board [7]. The accelerometer is installed in the middle of a cap.

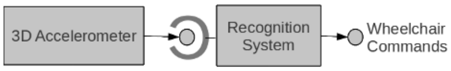

Fig. 1. System Block Diagram.

There are 4 different commands: left, right, forward, and backward. These commands are detected by tipping the head in a nodding action for the desired direction. Each command signal has a length of 2 seconds sampled with approximately 10Hz sampling frequency. The trigger signaling the beginning of a command is detected when a sudden change of acceleration with the magnitude of 0.05g (0.49m/s^2) is measured. When a trigger is detected, reference acceleration at time zero is stored. The following acceleration is normalized towards the reference. The 3-D acceleration data is stored in form of vectors of x-y-z.

Mitchell Yuwono is with Faculty of Engineering and Information Technology, University of Technology Sydney, Broadway, NSW 2011, Australia. (e-mail: Mitchell.Yuwono@student.uts.edu.au)

A.M. Ardi Handojoseno is with Faculty of Engineering and Information Technology, University of Technology, Sydney, Broadway, NSW 2011, Australia. (e-mail: AluysiusMariaArdi.Handojoseno@student.uts.edu.au)

Hung T. Nguyen is with Faculty of Engineering and Information Technology, University of Technology, Sydney, Broadway, NSW 2011, Australia (e-mail: Hung.Nguyen@uts.edu.au).

978-1-4244-4122-8/11/$26.00 ©2011 IEEE

III. Regrouping Particle Swarm Optimization Clustering

PSO was originally introduced by Kennedy and Eberhart in 1995 [8]. It has two base models: Local Best (*lbest*) PSO and Global Best (*gbest*) PSO. This paper will utilize the *gbest* PSO. Each particle in the swarm contains:

- x_i: the *current coordinate vector* of the particle,
- v_i: *current velocity vector* of the particle, and
- p_i: defines the *personal best coordinate vector*.

```
Algorithm $g$ = RegPSO_Cluster($z,N_c,N_p,k,c_1,c_2,\lambda,\varepsilon$)
$range(z) = \max(z) - \min(z)$                                        (1)
$v_{\max} = \lambda range(z)$                                         (2)
$g = K\_means(z, N_c)$                                                (3)
For each particle $i$, $1 \le i \le N_p$
    $x_i = K\_means(z, N_c)$                                          (4)
    $v_i = \vec{0}$                                                   (5)
    $p_i = x_i$                                                       (6)
End For
While stopping criterion is not satisfied
    $w(t+1) = kw(t)$                                                  (7)
    For each particle $i$, $1 \le i \le N_p$
        $v_i(t+1) = w(t)v_i(t) + c_1r_1(p_i(t) - x_i(t)) + c_2r_2(g(t) - x_i(t))$ (8)
        $x_i(t+1) = x_i(t) + v_i(t+1)$                                (9)
        $clamp(x_i, range(z))$                                        (10)
        $clamp(v_i, \pm v_{\max})$                                    (11)
        Calculate fitness of particle $x_i$ and update $p_i$ and $g$
        $f(x_i(t+1)) = \frac{\sum_{j=1}^{N_c} \sum \frac{d(z_p, x_{i,j}(t+1))}{\dim(C_j)}}{N_c}$ (12)
        If $f(x_i(t+1)) > f(p_i(t))$
            $p_i(t) = x_i(t+1))$                                      (13)
        End If
        If $f(x_i(t+1)) > f(g(t))$
            $g(t) = x_i(t+1))$                                        (14)
        End If
        Calculate distance of particle to global best
        $\delta_i = \frac{\|x_i(t+1) - g(t+1)\|}{\|range(z)\|}$       (15)
    End For
    If premature convergence criteria is satisfied, do regroup
    If $\max(\delta_i) < \varepsilon$                                 (16)
        For each particle $i$, $1 \le i \le N_p$
            $x_i = g + r \circ range(z) - \frac{range(z)}{2}$         (17)
        End For
    End If
End While
End Algorithm
```

Fig. 2. RegPSO cluster optimization algorithm

In the context of clustering, every particle in the swarm resembles a set of centroid vectors. Every particle is seeded with K-means (3), (4). The fitness of each particle can be calculated as the average of mean Euclidian distances between input data with the centroids (12) [8].

For each velocity and position update (8) (9), there exist random numbers r_1 and r_2, cognitive constant c_1, social constant c_2, and inertia weight w (7). Large w encourages exploration whereas small w encourages exploitation. x_i and v_i of each particle are clamped to prevent overflow. x_i is clamped inside the search space Ω (10). v_i is clamped in proportion λ of the range(Ω) (11) [10], [11].

Regrouping Particle Swarm Optimization (RegPSO) is used to remedy premature convergence and local minima problems [11]. A premature convergence is detected when the normalized swarm radius δ_{norm} is lower than the stagnation threshold ε = 1.1e-4 [11]. δ_{norm} can be defined as the ratio of the maximum Euclidian distance of any particle from its global best with the diameter of the search space(15)(16). Regroup results in re-initialization of particle positions centered around g (17). The RegPSO clustering algorithm can be seen in Fig.2, optimization process in Fig.3, and the resulting cluster in Fig.4.

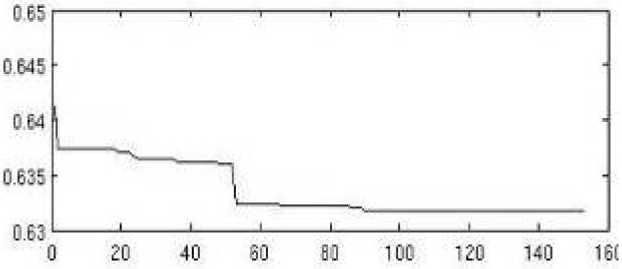

Fig.3. RegPSO cluster optimization process. *gbest* fitness graph vs. iterations (x-axis: number of iterations; y-axis: average Euclidian distances between input data and each centroids).

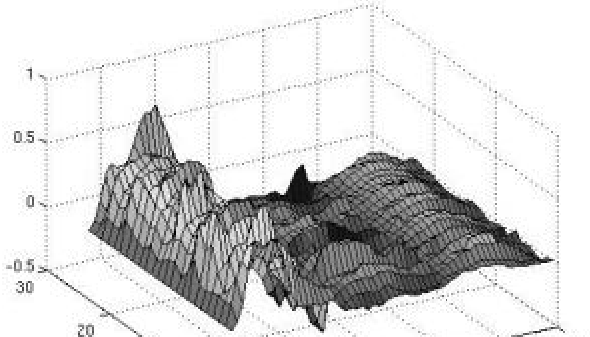

Fig. 4. Clustered 'left' command signals. The K-means seeded RegPSO algorithm appropriately clusters 24 input data. The Euclidian distance between the data in the cluster and its centroid is optimized up to 0.5627.

IV. Augmented Radial Basis Function Neural Networks

The ARBF-NN consists of two parts as is illustrated in Fig.5: The RBF layer and an MLP-NN augmentation layer. The RBF layer contains trained classifiers whose centroids are obtained from III. The output is the cluster membership rate. MLP-NN then translates the membership combinations to appropriate commands.

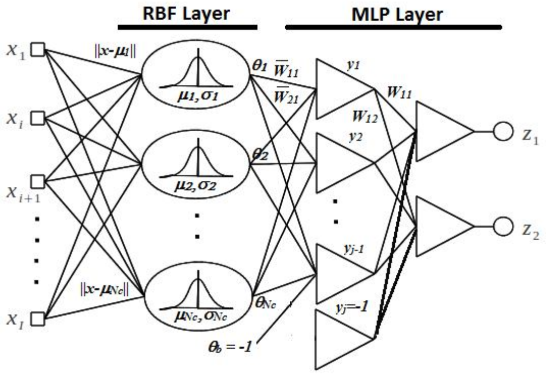

Fig.5. ARBF-NN Configuration

The RBF layer clusters the input information x into N_c groups. Each node in the layer uses a radial basis function. The radial basis function used is the Gaussian radial function (18). It can be described as a multi-dimensional Gaussian distribution with the center of μ and standard deviation of σ. Radius r^2 of the basis function is equal to $2\sigma^2$ [12]. Output of the RBF layer θ is a vector of cluster membership as is seen in Fig.6.

$$\theta_n(x) = e^{\frac{\|x-\mu_n\|}{2\sigma_n{}^2}} \quad n = \{1,2,\cdots,N_c\} \tag{18}$$

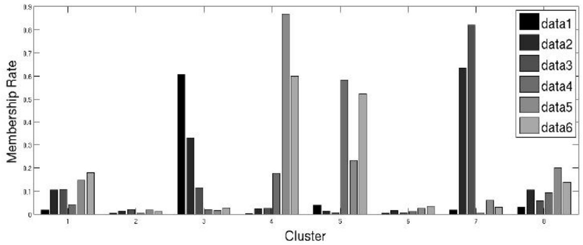

Fig. 6. Membership rate of input command signals. Data 1-3 are 'Left' command signals, data 4-6 are 'Right' command signals. X-axis shows the cluster numbers, y-axis shows the membership rate (0-1.0)

Both μ and r^2 are optimized with RegPSO. Radii optimization requires the connection of RBF-layer to a linear classifier such that it is configured as a standard RBF-NN. Each particle x_i resembles a vector of optimum radii candidate. Fitness function in (12) is changed as such: The output matrix of the RBF layer can be determined by calculating $\theta(z_p)_{2\sigma^2=x}$; Connection weight W can be optimized by performing least square between the target matrix d and RBF layer output matrix θ for every input data (19). The final fitness function can be seen in (20).

$$W = d / \theta \tag{19}$$

$$f(x) = \frac{1}{P}\sum_{p=1}^{P}\sqrt{\left(W\theta(z_p)_{2\sigma^2=x} - d_p\right)^2} \tag{20}$$

The purpose of the augmentation of an MLP layer after RBF layer is to negate the optimization limitation posed by linear classifiers in conventional RBF-NN [4].

V. Experimental Results and Discussions

A. Data Acquisition

Head movement training data was taken from six healthy adults using 3-axis accelerometer installed in the middle of a cap. 40 samples of command patterns for each person was collected and divided randomly into training data set (85%) and validation data set (15%).

Four set of test experiments were run in real time. The main J2SE program stores a log file of detections done by the system. This approach allows unbiased objective comparisons to be carried. The first experiment was done using a standard MLP network trained with Variable Learning Rate Backpropagation (GDX). The result of this training was used to test the head movement recognition system. The second and the third experiment were taken with similar steps using RBF and ARBF-NN configuration respectively. The fourth experiment was repeating the first three experiments by testing the system on four subjects that were not included in training or validation data (*outgroup*).

B. Parameters

The input layer consists of 60 neurons with 20 samples from each axis (x, y, and z). The RBF hidden layer consists of 20 neurons. The RBF hidden layer is connected to a linear classifier in case of standard RBF and to the input layer of an MLP network in case of ARBF. MLP hidden layer consists of 12 neurons as the result of trial and error for both ARBF and MLP configuration. Both the MLP input layer and hidden layer has an additional input bias neuron of value -1. All configurations have an output layer consisting of 4 neurons which relates to number of commands to be classified.

C. Results

As can be seen in Table I, II and III, specific for head movement recognition problem, ARBF shows accuracy advantage over MLP and RBF. There is a slight decrease in specificity as compared to RBF configuration with optimized centroids and radii. The ARBF configuration improves the sensitivity of an optimized RBF to a significant extent. Table IV shows improvements of accuracies until 8% from conventional RBF or MLP configurations to *ingroup* subjects. It is evident in the tables that sensitivity and specificity of the network are improved using the ARBF configuration. Performance of head movement system with ARBF configuration is relatively stable even when tested to *outgroup* subjects.

TABLE I.
Experiment I Confusion Matrix for Ingroup User Test Set
(MLP-NN Configuration)

		Predicted Classification			
Movement		Left	Right	Forward	Back
Actual Classification	Left	40	0	0	0
	Right	0	30	0	0
	Forward	0	4	36	0
	Backward	0	0	0	38
	Unknown	0	6	4	2
Accuracy (%)		90.0			

TABLE II.
Experiment II Confusion Matrix for Ingroup User Test Set
(RBF-NN Configuration)

		Predicted Classification			
Movement		Left	Right	Forward	Back
Actual Classification	Left	32	0	0	0
	Right	0	35	0	0
	Forward	0	0	37	0
	Backward	0	0	0	39
	Unknown	8	5	3	1
Accuracy (%)		89.4			

TABLE III.
Experiment III Confusion Matrix for Ingroup User Test Set
(ARBF-NN Configuration)

		Predicted Classification			
Movement		Left	Right	Forward	Back
Actual Classification	Left	39	0	0	0
	Right	0	39	1	0
	Forward	0	0	39	0
	Backward	0	1	0	40
	Unknown	1	0	0	0
Accuracy (%)		98.1			

TABLE IV.
Sensitivity and Specificity of Different Training Methods with RBF, MLP, and ARBF-NN Configuration

User and Methods	Sensitivity	Specificity
ingroup, MLP	90.0	99.17
ingroup, RBF	89.4	100
ingroup, ARBF	98.1	99.58
outgroup, MLP	85.00	96.25
outgroup, RBF	79.38	99.58
outgroup, ARBF	98.1	99.74

RBF network has the tendency to not extrapolate beyond known data and provides high stability over patterns as is depicted in Table IV. It enhances the safety which is critical for this application. On the other hand MLP network suffers from misclassifications. It, however, performs better in global generalization. [13]

ARBF network benefits from both networks strength and weakness. Having an RBF input layer stabilizes the system allowing it to have higher specificity by restricting global generalization. The MLP augmentation layer allows the network to be more tolerant than a standard RBF.

VI. Conclusion and Future Directions

Based on the experiment discussed on this paper, we concluded that the proposed ARBF-NN configuration improves the performance of a standard RBF-NN and MLP-NN for head movement recognition. We have also shown that RegPSO is a powerful optimization tool for both clustering and radii optimization.

ARBF-NN trained with current training data is able to recognize head movement pattern of outgroup users up to 98.1% sensitivity and 99.74% specificity. In order to further increase the performance of the proposed configuration, higher variability training samples from more individuals are required.

The success of the proposed ARBF-NN configuration to improve the robustness of real time head movement pattern recognition opens a possibility that this configuration may essentially increase the performance of RBF-NN for general pattern recognition problems. We are currently carrying further research using ARBF-NN for other pattern recognition problems to further exploit the capability of this configuration.

References

[1] L.M. King, H.T. Nguyen, and P.B. Taylor, “Hands-free Head-movement Gesture Recognition using Artificial Neural Networks and the Magnified Gradient Function”, in *Proc. of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, Shanghai, 2005, pp. 2063-2066.

[2] H.T. Nguyen, L.M. King, and G.Knight, “Real-Time Head Movement System and Embedded Linux Implementation for the Control of Power Wheelchair”, in *Proc of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology*, San Francisco, California, USA, 2004, pp. 4892-4895.

[3] D.A. Craig and H.T. Nguyen, “Wireless Real-Time Head-Movement System Using a Personal Digital Assistant (PDA) for Control of a Power Wheelchair”, in *Proc. of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, Shanghai, 2005, pp. 772-775.

[4] K.J. McGarry, S. Wermter, and J. MacIntyre, “Knowledge Extraction from Radial Basis Function Network and Multi-layer Perceptrons”, in *International Joint Conference on Neural Networks ‘99*, vol.4, Washington D.C., USA, 1999, pp.2494-2497.

[5] M.G.Passos, P.H.Silva, and H.C.C.Fernandes, “A RBF/MLP Modular Neural Network for Microwave Device Modeling”, in *Internatioanl Journal of Computer Science and Network Security*, vol.6 No.5A, 2006, pp.81-86.

[6] L.Ozyilmaz and T.Yildrim, “Diagnosis of Thyroid Disease Using Artificial Neural Network Methods” in *Proceedings of the 9th International Conference on NeInformation Processing (ICONIP’02)*, vol.4, pp.2033-2036.

[7] P. Lajšner and R. Kozub, *Using the MMA7360L ZSTAR2 Demo Board*, Freescale Semiconductor, November 2007.

[8] R. Eberhart and J. Kennedy, “A New Optimizer Using Particle Swarm Theory”, in *IEEE Sixth International Symposium on Micro Machine and Human Science*, 1995, pp. 39-43.

[9] D.W. Merwe and AP Engelhrecht, “Data Clustering using Particle Swarm Optimization” in *Congress on Evolutionary Computation*, vol.1, 2003, pp. 215 – 220.

[10] J. Kennedy, R.C. Eberhart, and Y. Shi, *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, CA, USA, 2001.

[11] G.I. Evers and M.B. Ghalia, “Regrouping Particle Swarm Optimization: A New Global Optimization Algorithm with Improved Performance Consistency Across Benchmarks”, in *International Conference on Systems, Man, and Cybernetics 2009*, San Antonio, TX, USA, 2009, pp.3901-3908.

[12] I.A. Nor, S. Harun, and A.H.M. Kassim, “Radial Basis Function Modelling of Streamflow Hydrograph”, in *Journal of Hydrologic Engineering*, vol.12, no.1, 2007, pp.113-123.

[13] T. Hill and P. Lewicki, *Statistics : Methods and Applications – A comprehensive reference for science, industries, and data mining,* StatSoft, Inc., Tulsa, OK, USA, 2006.