

## Research Article

# MIMO Lyapunov Theory-Based RBF Neural Classifier for Traffic Sign Recognition

King Hann Lim,<sup>1</sup> Kah Phooi Seng,<sup>2</sup> and Li-Minn Ang<sup>3</sup>

<sup>1</sup>Electrical and Computer Department, School of Engineering and Science, Curtin University, Sarawak Malaysia, CDT 250, 98009 Miri Sarawak, Malaysia

<sup>2</sup>School of Computer Technology, Sunway University, No. 5, Jalan Universiti, Bandar Sunway, Selangor Darul Ehsan, 46150 Petaling, Malaysia

<sup>3</sup>Centre for Communications Engineering Research, Edith Cowan University, Joondalup, WA 6027, Australia

Correspondence should be addressed to King Hann Lim, glkhann@curtin.edu.my

Received 27 October 2011; Revised 21 February 2012; Accepted 22 February 2012

Academic Editor: Toly Chen

Copyright © 2012 King Hann Lim et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Lyapunov theory-based radial basis function neural network (RBFNN) is developed for traffic sign recognition in this paper to perform multiple inputs multiple outputs (MIMO) classification. Multidimensional input is inserted into RBF nodes and these nodes are linked with multiple weights. An iterative weight adaptation scheme is hence designed with regards to the Lyapunov stability theory to obtain a set of optimum weights. In the design, the Lyapunov function has to be well selected to construct an energy space with a single global minimum. Weight gain is formed later to obey the Lyapunov stability theory. Detail analysis and discussion on the proposed classifier's properties are included in the paper. The performance comparisons between the proposed classifier and some existing conventional techniques are evaluated using traffic sign patterns. Simulation results reveal that our proposed system achieved better performance with lower number of training iterations.

## 1. Introduction

Traffic sign recognition is important in autonomous vehicular technology for the sake of identifying a sign functionality through visual information capturing via sensors. The usage of neural networks has become increasingly popular in traffic sign recognition recently to classify various kinds of traffic signs into a specific category [1–3]. The reason of applying neural networks in traffic sign recognition is that, they can incorporate both statistical and structural information to achieve better performance than a simple minimum distance classifier [4]. The adaptive learning capability and processing parallelism for complex problems have led to the rapid advancement of neural networks. Among all neural networks, *radial basis function neural network* (RBFNN) has been applied in many engineering applications with the following significant properties: (i) universal approximators [5]; (ii) simple topological structure

[6] which allows straightforward computation using a linearly weighted combination of single hidden-layer neurons. The learning characteristic of RBFNN is greatly related to the associative weights between hidden-output nodes. Therefore, an optimal algorithm is required to update the weights relative to an arbitrary training input.

Conventionally, the training process for RBFNN is mainly dependent on the optimization theory. The cost function of this network, for instance, the sum of squared errors or mean squared error between network's output and targeted input is firstly defined. It is followed by minimizing the cost function in weight parameter space to search for a set of optimal weights. These optimal weights which are acquired throughout network training process can be used to perform some unique tasks, such as pattern classification.

In order to obtain the optimum weights, a number of training algorithms have been developed for RBFNN. Due to the linear-weighted combiner, network's weights can be

determined using *least mean square* (LMS) and *recursive least square* (RLS) algorithms. However, these algorithms suffer from several drawbacks and limitations. The LMS is highly dependent on the autocorrelation function associated with the input signals and slow convergence. RLS, on the other hand, provides faster convergence but they depend on the implicit or explicit computation using the inverse of input signal's autocorrelation matrix. Matrix inversion implies not only a higher computational cost and it also leads to network instability issue [7]. Other gradient search-based training algorithms also suffer so-called local minima problem, that is, the optimization search may trap at a local minimum of the cost function in the weight space if a set of initial values are arbitrarily chosen. For example, the cost function has a fixed structure in the weight space after the expression of the cost function is chosen. The parameter update law is only a means to search for the global minimum and independent of the cost function in the weight space.

To overcome the aforementioned problems, the optimization techniques using the Lyapunov stability theory has been proposed in [8] for adaptive filtering. This theory is further adopted to the design of RBFNN [9] which has been first proposed in realization of *finite-impulse response* (FIR) and *infinite-impulse response* (IIR) adaptive filters for signal noise filtering. The Lyapunov theory-based RBFNN has been increasingly popular in adaptive filter due to its stability guarantee by Lyapunov stability theory and an energy-space construction with a global minimum [9]. However, only single output is designed in the Lyapunov theory-based RBFNN and, hence, it is not suitable for classification problem. In the meantime, the Lyapunov stability theory is also applied to *multilayered neural network* (MLNN) [10] for solving *multiple inputs multiple outputs* (MIMO) problems. With Taylor series expansion, all MLNN weights between input-output layers are rearranged into a linear configuration with an assumption made, that is, the input-output layer's weights adjustment is dependent on its corrective output error. Therefore, it leads to longer training time with a couple numbers of weight linkage and it prone to have more weights uncertainties. Nevertheless, the Lyapunov theory-based neural classifiers have offered the following advantages [9, 10]: (i) fast error convergence, (ii) guarantee of stability, (iii) insensitivity to initial condition, and (iv) construction of weight space with a global minimum if a proper Lyapunov function is selected.

In this paper, the notion of Lyapunov stability theory on RBFNN can be extended and modified to solve MIMO problems such as traffic sign recognition in order to obtain a fast and reliable classification system. To reveal the performance of proposed system, the application for traffic sign classification is used for further discussion and analysis. The performance of the proposed RBFNN will be compared with [10] and the differences for both methods will be stated in the later section. Experimental results show that the proposed method leads to faster error convergence rate and higher recognition rate as compared to the conventional techniques. This paper is organized as the following: Section 2 discusses about the fundamental theory of RBF neural classifier while Section 3 explains the

theoretical design of the Lyapunov theory-based training algorithm. Section 4 describes an overview of traffic sign detection and recognition on Malaysia's traffic sign database. Some simulation results along with the application of traffic sign recognition are shown in Section 5 and it is finally followed by conclusion and future works.

## 2. Radial Basis Function Neural Network

A typical three-layer RBFNN [11] is illustrated in Figure 1 for pattern recognition. Such a network implements an input-output mapping:  $\mathfrak{X}^n \rightarrow \mathfrak{X}^m$ , where  $n$  depicts the number of inputs and  $m$  depicts the number of outputs. There are  $u$  hidden nodes connecting in between the input-output layer. Assuming the first layer of input vector,  $\mathbf{X}^{(1)}$  is set to be  $\mathbf{P} \in \mathfrak{X}^n$ , where the input data is arranged into column vector as  $\mathbf{P} = [p_1, p_2, \dots, p_n]^T$ . The RBF centers are denoted as  $C_j \in \mathfrak{X}^n$  ( $1 \leq j \leq u$ ). Each RBF unit is defined as:

$$X_j^{(2)}(P) = \exp\left(-\frac{\|\mathbf{X}^{(1)} - C_j\|^2}{\sigma_j^2}\right), \quad (1)$$

where  $\|\cdot\|$  indicates the Euclidean norm on the input space while  $\sigma_j$  is the Gaussian width of the  $j$ th RBF unit. The vector generated after the RBF neurons is given as  $\mathbf{X}$  below:

$$\begin{bmatrix} \psi(\|\mathbf{X}^{(1)} - C_1\|^2) \\ \psi(\|\mathbf{X}^{(1)} - C_2\|^2) \\ \vdots \\ \psi(\|\mathbf{X}^{(1)} - C_u\|^2) \end{bmatrix} = \begin{bmatrix} X_1^{(2)} \\ X_2^{(2)} \\ \vdots \\ X_u^{(2)} \end{bmatrix} = \mathbf{X}, \quad (2)$$

where  $\psi(\cdot)$  is the radial basis function. Consider the hidden nodes are linearly mapped to the output with the  $u \times m$  weights matrix formed as below:

$$\begin{bmatrix} w_{1,1} & w_{2,1} & \cdots & w_{m,1} \\ w_{1,2} & w_{2,2} & \cdots & w_{m,2} \\ \vdots & & \ddots & \vdots \\ w_{1,u} & \cdots & \cdots & w_{m,u} \end{bmatrix} = \mathbf{W}, \quad (3)$$

RBF network establishes a linear function mapping in the output layer. By multiplying (2) and (3), the weighted hidden values are summed to be the output matrix,  $\mathbf{Y}$  is as follows:

$$\mathbf{Y} = \text{diag}(\mathbf{W}^T \mathbf{X}), \quad (4)$$

where

$$\mathbf{Y} = \text{diag}(y_1, y_2, \dots, y_m) = \begin{bmatrix} y_1 & \cdots & \cdots & 0 \\ 0 & y_2 & \cdots & 0 \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & y_m \end{bmatrix}, \quad (5)$$

and  $\text{diag}(a)$  is the  $N \times N$  diagonal matrix whose entries are the  $N$  elements of the vector  $a$ .

Two important parameters are associated with each RBF unit. They are RBF center ( $C_j$ ) and the Gaussian width ( $\sigma_j$ ).

Every center should well represent the corresponding subclass because the classification in RBFNN is mainly measured based on the distances between the input samples and the centers of each subclass. There are different strategies to select RBF centers [11]. These strategies can be classified into supervised [12, 13] and unsupervised algorithms [14–16]. Once centers are well selected, the network starts learning the training data with the weight updating scheme. The selection of center and radius is not the main concern in this paper. Therefore,  $k$ -mean cluster is applied to search for the RBF centers while  $p$ -nearest neighbor is used to measure the nearest radius for  $p$  consecutive centers in the network evaluations.

Learning algorithm plays an important role in updating the weight between hidden and output layer of RBFNN. Conventional RBF with LMS algorithm [17, 18] is trained using gradient descent method which finds the negative gradient on the error curve. It suffers from slow convergence and it is always trapped in the local minima instead of global minima. On the other hand, conventional RBF using RLS method [19] computes the inverse of autocorrelation matrix associated with the input data to update the system weights. However, the inversion of RLS algorithm gives instability of system convergence and increases computational cost.

### 3. Lyapunov Theory-Based RBFNN

The idea of Lyapunov theory-based RBF filter was initially developed in [9] for adaptive filtering. Lyapunov function of errors between targeted outputs and actual outputs are first defined. Network weights are then adjusted based on the Lyapunov stability theory, so that errors can asymptotically converge to zero. The selected Lyapunov function has a unique global minimum point in the state space. By properly choosing a weight update law in the Lyapunov sense, RBF outputs will be asymptotically converged to the target outputs. In this section, the design in [9] is adopted and modified to apply the Lyapunov theory-based RBF neural classifier for solving MIMO classification problem.

The input vector  $\mathbf{P}$  is fed into RBF nodes and hence passed to the output layer by weighted sum with the formulas depicted in (1)–(4). For the given desired response  $\hat{\mathbf{Y}}_k = \text{diag}(d_1, d_2, \dots, d_m) \in \mathfrak{R}^{m \times m}$  at discrete time  $k$ , the Lyapunov function is initially chosen as:

$$V_k = \|\mathbf{E}_k\|^2, \quad (6)$$

and  $\mathbf{E}_k$  is a posteriori error with  $m \times m$  diagonal matrix defined as below:

$$\mathbf{E}_k = \hat{\mathbf{Y}}_k - \mathbf{Y}_k = \hat{\mathbf{Y}}_k - \text{diag}(\mathbf{W}_k^T \mathbf{X}_k). \quad (7)$$

For the given hidden layer input  $\mathbf{X}_k$  and the desired output  $\hat{\mathbf{Y}}_k$ , the weight matrix  $\mathbf{W}_k$  is updated as follows:

$$\mathbf{W}_k = \mathbf{W}_{k-1} + \mathbf{g}_k^T \boldsymbol{\alpha}_k. \quad (8)$$

The adaptive gain is modified for MIMO such that  $\Delta V < 0$ :

$$\mathbf{g}_k = \left[ \mathbf{I}_{m \times m} - \kappa \boldsymbol{\alpha}_k^{-1} \mathbf{E}_{k-1} \right] \times \mathbf{1}_{m \times 1} \times \frac{\mathbf{X}_k^T}{\|\mathbf{X}_k\|^2}, \quad (9)$$

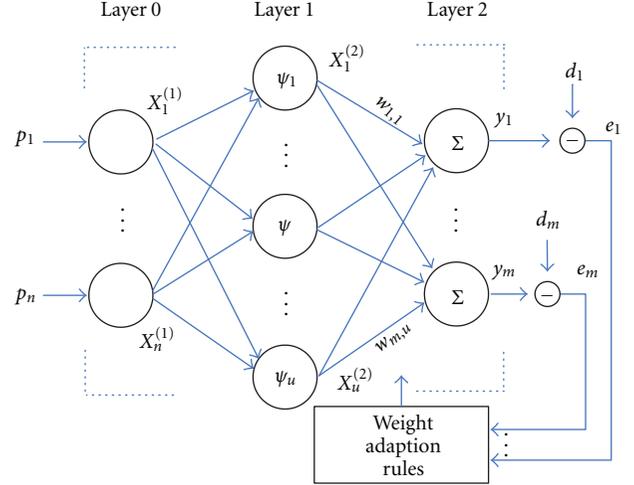


FIGURE 1: MIMO radial basis function network structure.

whereas  $\mathbf{I}_{m \times m}$  is identity matrix,  $\mathbf{1}_{m \times 1}$  is column vector of ones,  $\kappa$  is valid for the interval  $[0, 1]$ , and  $\boldsymbol{\alpha}_k$  is the priori estimation error which is defined as:

$$\boldsymbol{\alpha}_k = \hat{\mathbf{Y}}_k - \text{diag}(\mathbf{W}_{k-1}^T \mathbf{X}_k). \quad (10)$$

Then the error  $\|\mathbf{E}_k\|$  asymptotically converges to zero as the time  $k$  goes infinity.

The selection of Lyapunov function is important in constructing a new cost-function of the system. Regarding to the Lyapunov stability theory,  $V_k$  should be selected in the sense of Lyapunov that:  $\Delta V_k = V_k - V_{k-1} < 0$ . For  $m$ th output nodes RBFNN network,  $V_k$  is chosen to be (6) with the summation of squared errors. With the parameters predefined in expressions (7)–(10), it is proved in Appendix A that  $\Delta V_k$  has a negative value and the Lyapunov stability theory is fulfilled. Only when the parameter update law is chosen in the Lyapunov sense,  $V_k = \|\mathbf{E}_k\|^2$  is the Lyapunov function of RBFNN system, which has a unique global minimum.

As stated previously, the training error converges to zero asymptotically as time increases to infinity. For  $m$ th output nodes RBFNN, the classification error,  $\|\mathbf{E}_k\|$  is proved to be asymptotically approaching zero when the training time increases with the gain given in (9). The proof of error convergence is given in Appendix B. The error is bounded to a single convergent value. It is noted that the error convergence rate is dependent on the positive constant  $\kappa$ . For the faster error converges,  $\kappa$  should be remained as a small value in the range of  $0 \leq \kappa < 1$ .

To prevent the singularities, the expression (9) can be modified to:

$$\mathbf{g}_k = \left[ \mathbf{I}_{m \times m} - \kappa \boldsymbol{\alpha}_k^{-1} \mathbf{E}_{k-1} \right] \times \mathbf{1}_{m \times 1} \times \frac{\mathbf{X}_k^T}{\beta_1 + \|\mathbf{X}_k\|^2}, \quad (11)$$

where  $\beta_1$  is a small positive integer.

TABLE 1: Malaysia traffic sign classification in shape and color.

Color	Shape				
	Diamond	Triangle downward	Square/rectangle	Circle	Octagon
Blue	—	—	Information	Obligation	—
Red	—	Yield sign	—	Prohibition	Stop sign
Yellow	Warning	—	Warning	—	—
Orange	—	—	Construction	—	—
White	—	—	—	Speed limit	—

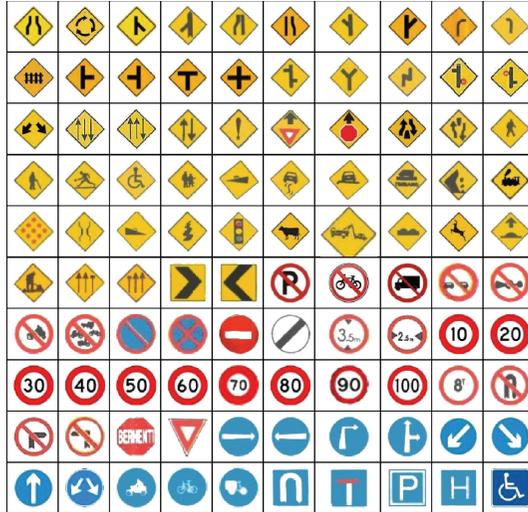


FIGURE 2: 100 classes of Malaysia traffic signs.

#### 4. Traffic Sign Detection and Recognition for Malaysia Traffic Sign Database

Traffic signs are important to alert driver of the current road situation. As demonstrated in Figure 2, 100 classes of traffic signs can often be found on Malaysia's roadside. The iconic traffic signs are acquired from the Malaysia's Highway Code Test booklet. They are designed in the standard geometrical shapes such as triangle, circle, octagon, rectangle, square, or diamond. The dominant colors that used for traffic signs are yellow, blue, red, orange, black, and white, which is greatly distinguishable from the natural scene. The message of warning, prohibition, guidance, construction and maintenance are represented by the specific color and shape as depicted in Table 1. They may contain a pictogram, a string of characters or both.

Traffic sign is detected based on color and shape information from a road scene. Possible sign is then extracted for further verification. Traffic sign can be verified by using its symmetrical property. Subsequently, the detected sign region is arranged into a column vector and this vector is inserted to the neural networks for classification. Some evaluations on traffic sign recognition using Malaysia's database with the proposed RBFNN and other neural classifiers will be discussed in the following section.

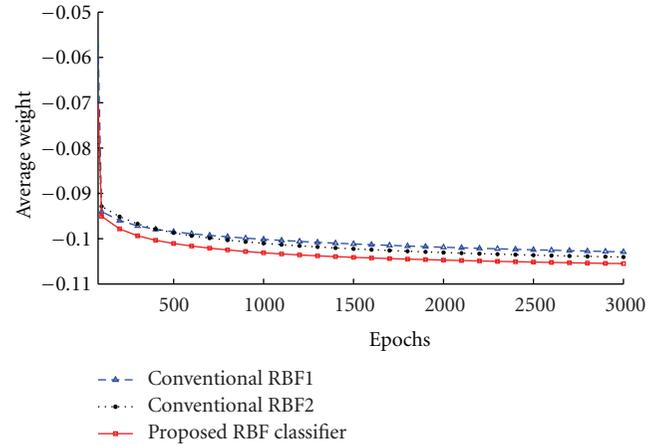


FIGURE 3: Weight convergence for different RBF classifiers.

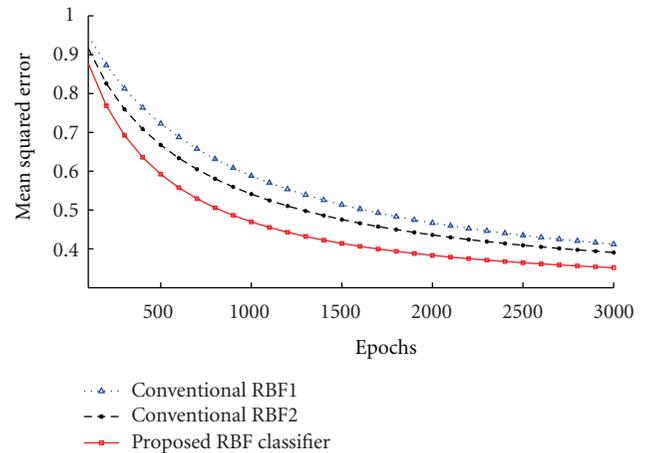


FIGURE 4: Error convergence on RBF networks.

#### 5. Experimental Results and Discussions

The Lyapunov theory-based RBFNN was developed in this paper to have multiple output classification and, therefore, traffic sign recognition was applied to be a MIMO problem throughout several experiments. In order to observe and evaluate the performance of proposed RBFNN, a basic structure of RBFNN was set up, where the output layer was assigned to classify 100 tasks. The task was to recognize any 100 signs randomly picked from Malaysia's database as displayed in the previous section. Each traffic sign subject

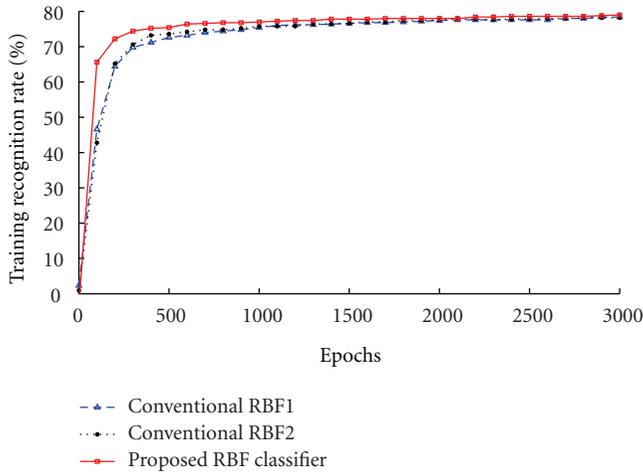


FIGURE 5: The training recognition rate in RBF networks versus epochs.

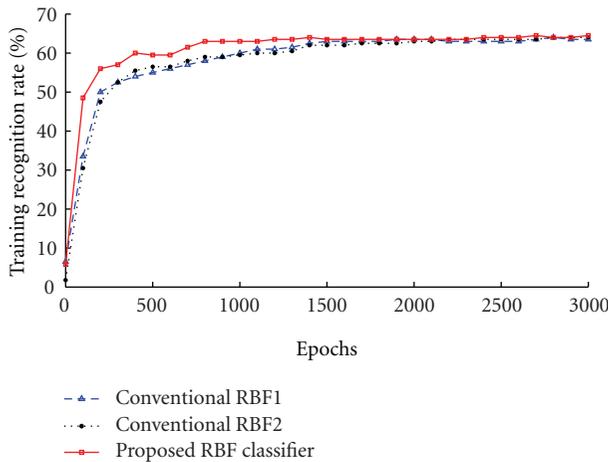


FIGURE 6: The testing recognition rate in RBF networks versus epochs.

contained five training images and two testing images respectively. All images were in the dimension of  $32 \times 32$ .

Three types of RBF classifiers with different learning schemes were used in the experiments, that is, (i) conventional RBF1 with LMS learning approach, (ii) conventional RBF2 with RLS technique, and (iii) Lyapunov theory-based RBF classifier. Network performance was compared in terms of weight and error convergence for these RBF systems. In addition, recognition rate was plotted along with the number of iterations for training and testing dataset. Lyapunov theory-based algorithm obtained a parameter,  $\kappa = 0.1$ , while LMS algorithm obtained a learning rate ( $\eta$ ), whereas the range was varied from 0.1 to 1.0. Meanwhile, the forgetting factor ( $\lambda$ ) of RLS algorithm was set to 0.1. During the experiments, the number of input neurons was set to be the image size and the number of output neurons was equal to the number of classes to be identified. The hidden layer of RBF nodes obtained the same size as the input layer. Two

initial conditions were set as (i) initial weights  $\mathbf{W}_0$  were set to be some random small values and (ii) initial posterior error,  $\mathbf{E}_0$  was set to be  $0.01 \times \mathbf{I}_{m \times m}$ , where  $m = 100$ . In the experiment, an epoch represented an efficient time used to train the entire set of input data. Hence, each epoch updated the weights once for the whole training dataset until the occurrence of weight convergence and this was referring to batch learning scheme.

Weight convergence was crucial in neural network learning characteristic. The weights for all RBF classifiers were converged at equilibrium with a random set of initial weights. Figure 3 showed a weight convergence plot, where  $x$ -axis denoted the number of epochs while  $y$ -axis denoted the average weight values connected from hidden layer to output layer of RBF network. At around 250 epochs, proposed RBF classifier started converging at  $-0.1$ . As time extended to infinity, only minor weight increment was added along with the continuous epoch. However, conventional RBF classifiers reached  $-0.1$  equilibrium point at 1000 epochs, which took longer time than the Lyapunov theory-based RBF classifier. Therefore, the weight of proposed RBF classifier was converged faster than other conventional RBF classifiers.

As demonstrated in Figure 4, an error curve was plotted and it was exponentially decreased to the minimum. At 2000 epochs, the proposed RBF classifier reached 0.4 of mean squared error. For other conventional RBFNNs, they took longer period to achieve lower error rate as shown in Figure 4. Hence, the error convergence rate of proposed RBF training scheme was converged faster than other networks. However, the error value was not converged to zero yet due to some reasons, whereby this issue was caused by the redundancy information of input data. Moreover, similarity of subject pattern would confuse the classification of neural network as you could observe in Figure 2. The complexity of the input data would degrade recognition rate in classification. Hence, all neural classifiers' error was still remained at around 0.4, but the proposed neural classifier obtained lower error curve compared to the other networks. To reduce the classification error, some processing techniques could be applied to the input data before passing to the neural network for training process. In addition, traffic signs could first be classified into color and shape before they were recognized using neural network [4]. In this paper, the main concern was to focus on the property of RBF training schemes instead of traffic sign recognition, in which it was an application example for the proposed network discussion.

To investigate the performance of RBF classifiers, training and testing data were fed into system and an average recognition rate was calculated. Without any feature extraction method, the original images were tested with the RBF classifiers. The system performance was compared with conventional RBFNNs using different weight updating schemes. As demonstrated in Figure 5, the Lyapunov theory-based RBFNN obtained faster training speed which was within 500 epochs. Meanwhile, it achieved higher training recognition rate than other networks. Similar to Figure 6, the proposed RBF network obtained high recognition rate at 800 epochs while other training algorithms achieved maximum recognition rate at 1500 epochs. This implied that Lyapunov

theory-based RBFNN could obtain high recognition rate in a smaller epoch's number. By comparing the network recognition rate, conventional RBF1 achieved 63.50%, while conventional RBF2 achieved 64% of recognition rate. However, the testing recognition rate of the proposed classifier was slightly increased with the rate of 64.50%. As it was reported in [12], high dimensional input data would cause low recognition rate of neural network because high dimensional input data and network complexity would need a large set training samples. Therefore, features extraction technique such as *Principle Component Analysis* (PCA) could be used to reduce the image dimension in order to further increase the recognition rate.

Besides that, MLNN was also used to classify multidimensional patterns based on Lyapunov stability theory as it was developed in [10]. In order to test the performance between the Lyapunov theory-based RBFNN and MLNN, the number of hidden nodes was fixed to be 100 and the number of epochs was set to be 3000 for both networks. The constant parameter  $\kappa$  is set to be 0.1 for both the Lyapunov-theory based MLNN and RBFNN with similar training and testing images datasets as employed in the RBFNN experiments. An averaged recognition rate was recorded for the comparison.

As reported in Table 2, the Lyapunov theory-based RBFNN achieved better classification performance than [10]. Although both methods employ the Lyapunov stability theory as the basic design for training algorithm, there were some main differences between them. First, the structure for two networks was much distinguished from input to output layers. MLNN contained more weights linkage between input to hidden layer and hidden to output layer. Hence, the weights of MLNN were required to be linearized using Taylor series expansion before performing training process. With this MLNN configuration, it contributed to more uncertainties on the weight and error convergence compared to the linear topology of RBFNN. Second, the energy function of training error for MLNN was constructed for each output node where it contained  $t$ th number of output nodes. Unlike to [10], the energy function for the Lyapunov theory based RBFNN was designed for all output neurons in a matrix form and it provided a valid derivation in the appendices. Finally, the performance for the Lyapunov theory-based RBFNN achieved better training and testing recognition rate than [10] in the pattern recognition.

## 6. Conclusion

This paper has presented Lyapunov theory-based weights updating algorithm for RBFNN. The weight adaptation scheme is designed based on the Lyapunov stability theory and iteratively updated the RBFNN weight. The Lyapunov theory-based RBFNN is extended for traffic sign recognition as a MIMO problem. Simulation results have shown that our proposed system achieved faster training speed, as well as higher recognition rate. The recognition rate can be further improved by applying input dimensionality reduction to remove the information redundancy. The research on the optimization using Lyapunov stability theory is still at its

early stages, and many investigations of Lyapunov theory-based neural classifier will be conducted to improve the network efficiency and robustness. Future investigation on different Lyapunov functions and different weights updating laws is needed to further improve the performance.

## Appendices

### A. Proof of Lyapunov Stability Theory

The discrete form of  $\Delta V_k$  is given as:

$$\begin{aligned} \Delta V_k &= \|\mathbf{E}_k\|^2 - \|\mathbf{E}_{k-1}\|^2 \\ &= \left\| \hat{\mathbf{Y}}_k - \text{diag}(\mathbf{W}_k^T \mathbf{X}_k) \right\|^2 - \|\mathbf{E}_{k-1}\|^2 \\ &= \left\| \boldsymbol{\alpha}_k - \boldsymbol{\alpha}_k^T \mathbf{g}_k \mathbf{X}_k \right\|^2 - \|\mathbf{E}_{k-1}\|^2 \\ &= \|\kappa \mathbf{E}_{k-1}\|^2 - \|\mathbf{E}_{k-1}\|^2 \\ &= \kappa^2 \|\mathbf{E}_{k-1}\|^2 - \|\mathbf{E}_{k-1}\|^2 \\ &= -(1 - \kappa^2) \|\mathbf{E}_{k-1}\|^2 < 0, \end{aligned} \quad (\text{A.1})$$

where  $0 \leq \kappa < 1$ .

### B. Proof of Error Convergence

$$\begin{aligned} \mathbf{E}_k &= \hat{\mathbf{Y}}_k - \mathbf{Y}_k \\ &= \hat{\mathbf{Y}}_k - \text{diag}(\mathbf{W}_k^T \mathbf{X}_k) \\ &= \hat{\mathbf{Y}}_k - \text{diag}\left[ (\mathbf{W}_{k-1}^T + \boldsymbol{\alpha}_k^T \mathbf{g}_k) \mathbf{X}_k \right] \\ &= \hat{\mathbf{Y}}_k - \text{diag}(\mathbf{W}_{k-1}^T \mathbf{X}_k) - \text{diag}(\boldsymbol{\alpha}_k^T \mathbf{g}_k \mathbf{X}_k) \\ &= \boldsymbol{\alpha}_k - \text{diag}\left\{ \boldsymbol{\alpha}_k^T \left[ (\mathbf{I}_{m \times m} - \kappa \boldsymbol{\alpha}_k^{-1} \mathbf{E}_{k-1}) \right. \right. \\ &\quad \left. \left. \times \mathbf{1}_{m \times 1} \times \frac{\mathbf{X}_k^T}{\|\mathbf{X}_k\|^2} \right] \mathbf{X}_k \right\}. \end{aligned} \quad (\text{B.1})$$

Since  $\boldsymbol{\alpha}_k$  is a symmetric matrix,  $\boldsymbol{\alpha}_k^T = \boldsymbol{\alpha}_k$ ,

$$\mathbf{E}_k = \boldsymbol{\alpha}_k - \text{diag}[(\boldsymbol{\alpha}_k - \kappa \mathbf{E}_{k-1}) \times \mathbf{1}_{m \times 1}]. \quad (\text{B.2})$$

Due to  $(\boldsymbol{\alpha}_k - \kappa \mathbf{E}_{k-1}) \in \mathfrak{R}^{m \times m}$ ,  $\text{diag}(\cdot)$  and  $\mathbf{1}_{m \times 1}$  can be canceled off from the equation:

$$\begin{aligned} \mathbf{E}_k &= \boldsymbol{\alpha}_k - \boldsymbol{\alpha}_k + \kappa \mathbf{E}_{k-1} \\ &= \kappa \mathbf{E}_{k-1} \\ \therefore \|\mathbf{E}_1\|^2 &= \kappa \|\mathbf{E}_0\|^2 \\ \|\mathbf{E}_2\|^2 &= \kappa \|\mathbf{E}_1\|^2 = \kappa^2 \|\mathbf{E}_0\|^2 \\ &\vdots \\ \|\mathbf{E}_k\|^2 &= \lim_{k \rightarrow \infty} \kappa^k \|\mathbf{E}_0\|^2 \approx 0, \end{aligned} \quad (\text{B.3})$$

where  $\|\mathbf{E}_0\|^2$  is an  $m \times m$  diagonal matrix with small real integer and  $0 \leq \kappa < 1$ .

TABLE 2: Performance comparison for the Lyapunov theory-based neural networks.

Method	Hidden nodes	Epochs	Training recognition (%)	Testing recognition (%)
Proposed RBF NN	100	3000	79.00	64.50
MLNN [10]	100	3000	62.80	56.00

## References

- [1] Y. Y. Nguwi and A. Z. Kouzani, "Detection and classification of road signs in natural environments," *Neural Computing and Applications*, vol. 17, no. 3, pp. 265–289, 2008.
- [2] Y. Shao, Q. Chen, and H. Jiang, "RBF neural network based on particle swarm optimization," in *Proceedings of the 7th International Conference on Advances in Neural Networks (ISNN '10)*, L. Zhang et al., Ed., vol. 6063, pp. 169–176, Springer, Shanghai, China, 2010.
- [3] G. A. P. Coronado, M. R. Muñoz, J. M. Armingol et al., "Road sign recognition for automatic inventory systems," in *Proceedings of the 18th International Conference on Systems, Signals, and Image Processing (IWSSIP '11)*, pp. 63–66, 2011.
- [4] K. H. Lim, K. P. Seng, and L.-M. Ang, "Improved traffic sign recognition," in *Proceedings of the International Conference on Embedded Systems and Intelligent Technology (ICESIT '10)*, Chiang Mai, Thailand, 2010.
- [5] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Computation*, vol. 3, no. 2, pp. 246–257, 1991.
- [6] S. Lee and R. M. Kil, "A gaussian potential function network with hierarchically self-organizing learning," *Neural Networks*, vol. 4, no. 2, pp. 207–224, 1991.
- [7] M. S. Mueller, "Least-squares algorithms for adaptive equalizers," *The Bell System Technical Journal*, vol. 60, no. 8, pp. 1905–1925, 1981.
- [8] Z. H. Man et al., "Design of robust adaptive filters using Lyapunov stability theory," in *Proceedings of the IEEE Transactions on Circuits & Systems II: Express Briefs*, 2004.
- [9] K. P. Seng, Z. Man, and H. R. Wu, "Lyapunov-theory-based radial basis function networks for adaptive filtering," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 8, pp. 1215–1220, 2002.
- [10] K. H. Lim, K. P. Seng, L. M. Ang, and S. W. Chin, "Lyapunov theory-based multilayered neural network," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, no. 4, pp. 305–309, 2009.
- [11] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 1994.
- [12] M. J. Er, S. Wu, J. Lu, and H. L. Toh, "Face recognition with radial basis function (RBF) neural networks," *IEEE Transactions on Neural Networks*, vol. 13, no. 3, pp. 697–710, 2002.
- [13] M. J. Er, W. Chen, and S. Wu, "High-speed face recognition based on discrete cosine transform and rbf neural networks," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 679–691, 2005.
- [14] J. A. Leonard and M. A. Kramer, "Radial basis function networks for classifying process faults," *IEEE Control Systems Magazine*, vol. 11, no. 3, pp. 31–38, 1991.
- [15] M. John and J. D. Christian, "Fast learning in networks of locally-tuned processing units," *Neural Computation*, vol. 1, no. 2, pp. 281–294, 1989.
- [16] F. Yang and M. Paindavoine, "Implementation of an rbf neural network on embedded systems: real-time face tracking and identity verification," *IEEE Transactions on Neural Networks*, vol. 14, no. 5, pp. 1162–1175, 2003.
- [17] M. Kishan, *Elements of Artificial Neural Networks*, MIT Press, 1997.
- [18] V. Espinosa-Duro, "Biometric identification system using a radial basis network," in *Proceedings of the 34th IEEE Annual International Carnahan Conference on Security Technology*, pp. 47–51, 2000.
- [19] M. Birgmeier, "Fully kalman-trained radial basis function network for nonlinear speech modeling," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 259–264, December 1995.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

