

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Vehicle Recognition System Using Singular Value Decomposition and Extreme Learning Machine

Zuraidi Saad, Muhammad Khusairi Osman,
Iza Sazanita Isa, Saodah Omar, Sopiah Ishak¹,
Khairul Azman Ahmad and Rozan Boudville

Faculty of Electrical Engineering &

¹Department of Computer Science and Mathematic,

Universiti Teknologi MARA

Malaysia

1. Introduction

The purpose of this research is to develop a system that is able to recognize and classify a variety of vehicles using image processing and artificial neural network. In order to perform the recognition, first, all the images containing the vehicles are required to go through several images processing technique such as thresholding, histogram equalization and edge detection before obtaining the desired dataset for classification process. Then, the vehicle images are converted into data using singular value decomposition (SVD) extraction method and the data are used as an input for training process in the classification phase. A Single Layer Feedforward (SLFN) network trained by Extreme Learning Machine (ELM) algorithm is chosen to perform the recognition and classification. The network is evaluated in terms of classification accuracy, training time and optimum structure of the network. Then, the recognition performance using the ELM training algorithm is compared with the standard Levenberg Marquardt (LM) algorithm.

2. Related study

Extreme learning machine (ELM) was proposed in Huang, *et al.* (2004) to provide the best generalization performance at extremely fast learning speed. Compared to ELM, the traditional implementations shown that the learning speed of Feedforward neural networks is in general far slower than required and it has been a major bottleneck in their applications for past decades. These are due to the slow gradient-based learning algorithms are extensively used to train neural networks, and all the parameters of the networks are tuned iteratively by using such learning algorithms. There are many researches have been conducted to compare the performance of ELM training algorithm for training a SLFN network with the other types of neural network training algorithms. Proposed research by Huang and Siew (2004) has extended to single layer feedforward (SLFNs) networks with radial basis function kernels of RBF networks with the implementation of ELM learning

algorithm to easily achieve good generalization performance at extremely fast learning speed. This method allows the centers and impact widths of the RBF kernels to be randomly generated and the output weights to be simply analytically calculated instead of iteratively tuned. The experimental results show that the ELM algorithm for RBF networks is able to complete learning at extremely fast speed and produce generalization performance almost similar to SVM algorithm in function approximation and classification problems. Wang and Huang (2005) have evaluated the performance of training the ELM algorithm and the Backpropagation (BP) training algorithms to classify an identified protein sequence and unseen protein sequence in feature patterns extraction of biological data. The study has indicated that the ELM training algorithm needed up to four orders of magnitude less training time as compared to BP algorithm. The ELM algorithm has given better classification accuracy performance as compared to BP algorithm. Moreover, the ELM does not have any control parameters to be manually tuned and hence can be implemented easily. Evolutionary ELM (E-ELM) algorithm has been proposed by Zhu *et al.* (2005) as an extended research from Huang and Siew (2004). The hybrid learning algorithm of E-ELM algorithm is preferably considered since the ELM algorithm may need higher number of hidden neurons due to the random determination of the input weights and hidden biases. The E-ELM algorithm uses the differential evolutionary algorithm to select the input weights and Moore-Penrose (MP) generalized inverse to analytically determine the output weights. This approach achieved a good generalization performance with much more compact networks as compared to other algorithms including BP, GALs (Ghosh and Verma, 2002) and the original ELM. In the conjunction of the above researches, this study is conducted to develop a different approach for classification task which relating between image processing method and artificial neural network. Liang *et al.* (2006) has shown that the ELM training algorithm performance needs an order of magnitude less training time for the Support Vector Machines (SVM) and two orders of magnitude less training time for the BP algorithm. However, classification accuracy performance of ELM algorithm is similar as the SVMs and BP algorithms. From the study, it also shown that classification accuracies can be improved by smoothing of classifiers' outputs. This research has been implemented to classify mental tasks from EEG signals to provide communication and control capabilities to people with severe or complete motor paralysis. Terrain reconstruction in path planning problem for supporting multiresolution terrain access has led to the study by Yeu *et al.* (2006). The ELM training algorithm has been implemented to speed up the rate for the network learns a priori available maps. From the results, it is shown that the ELM algorithm used during the query stage has performed better than BP, Delaunay Triangle (DT) and SVMs. Furthermore, the ELM training algorithm utilized far less memory for queries on large maps as compared to DT to achieve the same levels of MSE errors. Zhang *et al.* (2007) has quoted that the ELM is able to avoid the problems of local minima, improper learning rate and overfitting commonly faced by iterative learning methods and completes the training very fast. The research has been conducted to classify multcategory cancer based on microarray data sets of cancer diagnosis. The ELM algorithm classification accuracies have been compared with several training algorithms that are BP, Linder's SANN and SVMs of SVM-OVO and Ramaswamy's SVM-OVA. From the study results, it is indicated that when the number of categories for the classification task is large, the ELM algorithm achieves higher classification accuracy than the other algorithms with less training time and a smaller network structure. It can also be seen that ELM achieves better and more balanced classification for individual categories as well.

3. The proposed recognition system

The methodology used to develop the vehicle recognition system includes image acquisition, image processing, image extraction, image training and image testing using a SLFN network trained by the ELM and LM algorithm. To determine the suitability of the SLFN network in recognizing the images, it needs to go through training and testing phase. The training phase is chosen to be 96 dataset and 119 dataset for the testing phase. Once the network has learned the information in the training set and has converged, the test data is applied to the network for verification. The sigmoidal function is used for the hidden node activation function, for both the SLFN trained by the ELM and the BP training algorithms.

3.1 Image acquisition

Image acquisition device is set up as shown as in Figure 1 to obtain the best result in capturing vehicle images. The camera is placed above the bridge for a wider and better capturing area. A set of 119 data sample were captured in video format includes 52 images of motorcycle, 19 images of bus and 48 images of lorry. The image samples are then edited using Gretech Online Movie (GOM) Player to obtain the specific region and format needed for the training purposes. The Gom Player is able to play the majority of media files without the need to obtain a codec as well as play some broken media files. These will gives advantages over the traditional player, like Windows Media Player. Figure 2 shows samples images of motorcycle, bus and lorry that is used as data set in this research.

3.2 Image processing

Image processing is one of the major parts in this research due to the function of the images itself as the input for the training and testing process. All the possible noise, background or any other unwanted data in the images are removed to gain a stable system with high accuracy. In this process, firstly the original input image is cropped to create an interactive Crop Image tool associated with the image displayed in the current figure, called the target image which forms the input to the recognition system by using MATLAB. Sample of images after cropping process are shown in Figure 3. Figure 4 shows the images after converted to gray scale format. Next, the process of thresholding is applied to remove the background as illustrates in Figure 5. The qualities of the images are then enhanced by applying histogram equalization technique as shown in Figure 6.

The edge detection block by using Canny operator finds edges by looking for the local maxima of the gradient of the input image. It calculates the gradient using the derivative of

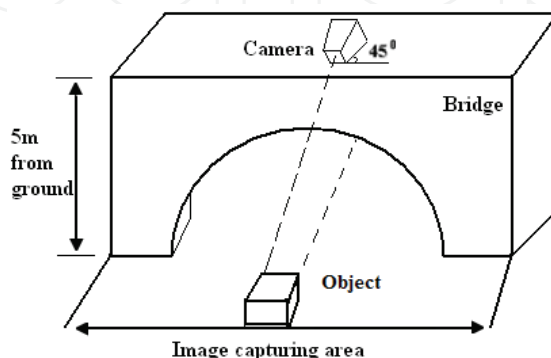


Fig. 1. Image acquisition set up



Fig. 2. Original input images



Fig. 3. The cropped images



Fig. 4. Gray image



Fig. 5. Thresholded images

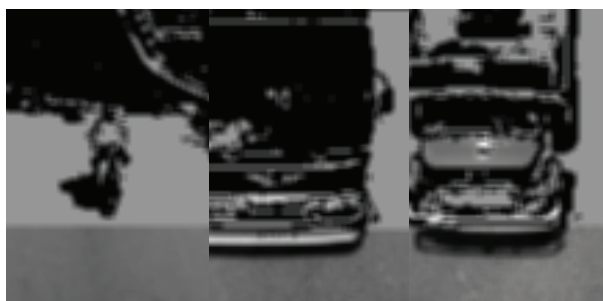


Fig. 6. Vehicle images after applying histogram equalization



Fig. 7. Vehicle images after applying Canny edge detection

the Gaussian filter. The Canny method uses two thresholds to detect strong and weak edges. It includes the weak edges in the output only if they are connected to strong edges (Saad *et al.* 2007). As a result, the method is more robust to noise, and more likely to detect true weak edges as shown in Figure 7.

3.3 Singular value decomposition

Singular Value Decomposition (SVD) is a factorization technique for rectangular matrices largely used in signal processing and pattern recognition. The method is applied to extract all of the images into a data set that can be as the input for neural network training and testing processes. The purpose of singular value decomposition is to reduce a dataset containing a large number of values to a dataset containing significantly fewer values, but which still contains a large fraction of the variability present in the original data.

A non-square data matrix A of size $m \times n$ with $m > n$ can be factorized into three matrices U , S , and V using singular value decomposition as shown in equation above. Here U is an $m \times m$ matrix, S is a $m \times n$ matrix and V is an $n \times n$ matrix. S is the diagonal matrix containing all of the non-negative singular values of the original data matrix listed in descending order. U and V are orthogonal square matrices representing the left and right singular vectors for the data matrix. U represents the row space and the transpose of V represents the column space (Cao 2007).

$$A = USV^T \tag{1}$$

where matrix U is an $m \times m$ orthogonal matrix

$$U = [u_1, u_2, \dots, u_r, u_{r+1}, \dots, u_m] \tag{2}$$

Column vectors u_i , for $i = 1, 2, \dots, m$, form an orthogonal set:

$$u_i^T u_j = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (3)$$

And matrix V is an $n \times n$ orthogonal matrix

$$V = [v_1, v_2, \dots, v_r, v_{r+1}, \dots, v_n] \quad (4)$$

Column vectors v_i for $i = 1, 2, \dots, n$, form an orthogonal set:

$$v_i^T v_j = \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (5)$$

Here, S is an $m \times n$ diagonal matrix with singular values (SV) on the diagonal. The matrix S can be showed in Equation 6.

$$s = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_r & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & \sigma_{r+1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & \sigma_n \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix} \quad (6)$$

For $i = 1, 2, \dots, n$, σ_i are called Singular Values (SV) of matrix A . It can be proved that:

$$\begin{aligned} \sigma_1 &\geq \sigma_2 \geq \dots \geq \sigma_r > 0, \\ \sigma_{r+1} &= \sigma_{r+2} = \dots = \sigma_n = 0 \end{aligned} \quad (7)$$

For $i = 1, 2, \dots, n$, σ_i are called Singular Values (SV) of matrix A . The v_i 's and u_i 's are called right and left singular-vectors of A (Cao 2007)

3.4 Multilayered perceptron network and Levenberg-Marquardt training algorithm

There are many different types of new training algorithms have been proposed. Traditionally we selected Levenberg-Marquardt (LM) training algorithm as because it is the fastest training speed on the same precision basis. According to Saodah *et al.* 2010, it shown that the LM learning algorithm has performed higher classification accuracy as compared to the other traditional learning algorithms. The LM method is an approximation of the Gauss-Newton technique, which generally provides faster learning rate than the back propagation that is based on the steepest decent technique. The learning is equivalent to finding a multidimensional function that provides a best fit to the training data, with the criterion for "best fit" being measured in some statistical sense. The recognition performance of the MLP network will highly depend on the structure of the network and the training algorithm. In the current study, the LM algorithm has been selected to train the network. It has been shown that the algorithm has much better learning rate than the famous back propagation

algorithm (Hagan and Menhaj 1994). The MLP with one single hidden layer, or also called as the Single Layer Feedforward Network (SLFN) network consists of three separate layers is shown in Figure 8. The input layer is the set of source nodes. The second layer is a hidden layer of high dimension. The output layer gives the response of the network to the activation patterns applied to the input layer (Mashor 2000).

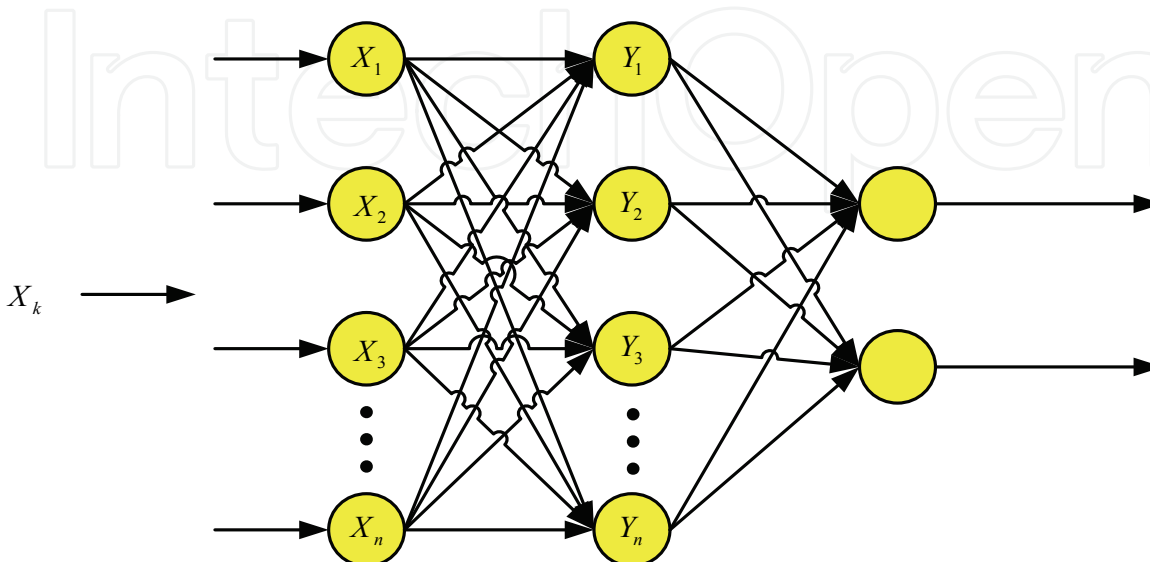


Fig. 8. Architecture of a SLFN network

The number of nodes in the input, hidden and output layers will determined the network structure. Furthermore, the hidden and output nodes have activation function that will also influence the network performance. The best network structure is normally problem dependent, hence structure analysis has to be carried out to identify the optimum structure. In the current study, the numbers of input and output nodes were fixed at 48 and 3 respectively, since the images have been divided into 48 segments and the target outputs are 3 classes of images. Therefore, only the number of hidden nodes and activation functions need to be determined. The percentage of classification performance will be used to judge the network performance to perform vehicle recognition. For this analysis the vehicle images without noise were used to determine the structure of the network. The analysis is used to determine number of hidden node, learning rate and sufficient training of epoch (Mashor 2000) & (Mashor 2004).

The Levenberg-Marquardt algorithm was designed to approach second-order training speed without having to compute the Hessian matrix. When the performance function has the form of a sum of squares (as is typical in training feedforward networks), then the Hessian matrix is based on :

$$H = J^T J \tag{8}$$

And the gradient can be computed as:

$$g = J^T e \tag{9}$$

where J is the Jacobian matrix that contains the first derivatives of the network errors with respect to the weights and biases, and e is a vector of network errors. The Jacobian matrix can be computed through a standard back propagation technique that is much less complex than computing the Hessian matrix. The Levenberg-Marquardt algorithm uses this approximation, μ to the Hessian matrix in the following Newton-like update according to Equation 10:

$$X_{k+1} = X_k - [J^T J + \mu I]^{-1} J^T e \quad (10)$$

A sigmoid function which is given by (11), is a mathematical function that produces a sigmoid curve as S shape and smoothes the transition between the input, t and the output $P(t)$. It is a real-valued and differentiable, having either a non-negative or non-positive first derivative and exactly one inflection point.

$$P(t) = \frac{1}{1 + e^{-t}} \quad (11)$$

3.5 ELM based training algorithm

Liang *et al.* (2006) has showing the ability of the SLFN network to fix the network connection at one layer with the weights between input neurons and hidden neurons. The same goes to the output neurons where there is fix network connection with weights between hidden neurons and output neurons. However, the algorithm was unable to adjust the weights on both layers simultaneously since there is no gain provided. Based on this work, Huang *et al.* (2006) have proposed a new learning algorithm referred to as Extreme Learning Machine (ELM). ELM is a learning algorithm that is derived based on some continuous probability density function. Consequently the ELM is designed to be randomly chooses and fixes the weights between input neurons and hidden neurons, and then analytically determines the weights between hidden neurons and output neurons of the SLFN.

For N arbitrary distinct samples (x_i, t_i) , where $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$ and $t_i = [t_{i1}, t_{i2}, \dots, t_{i3}]^T \in R^m$, standard SLFNs with N hidden nodes and activation function $g(x)$ are mathematically modelled as:

$$\sum_{i=1}^N \beta_i g_i(x_j) = \sum_{i=1}^N \beta_i g(w_i \cdot x_j + b_i) = o_j, \quad j=1, \dots, N, \quad (12)$$

where $w_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ is the weight vector connecting the i^{th} hidden node and the input nodes, $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ is the weight vector connecting the i^{th} hidden node and the output nodes, and b_i is the threshold of the i^{th} hidden node. $w_i \cdot x_i$ denotes the inner product of w_j and x_j .

The above N equations can be written compactly as:

$$H\beta = T, \quad (13)$$

Where

$$H(w_1, \dots, w_N, b_1, \dots, b_N, x_1, \dots, x_n) = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \dots & g(w_N \cdot x_1 + b_N) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_N + b_1) & \dots & g(w_N \cdot x_N + b_N) \end{bmatrix}_{N \times N}, \quad (14)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_N^T \end{bmatrix} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix} \quad (15)$$

Usually H is called the hidden layer output matrix of the neural network and the i th column of H represented the i th hidden node output with respect to inputs $x_1; x_2; \dots; x_N$.

3.6 ELM learning algorithm

In order to train the arbitrary function of neural network with zero training error, Baum (1988) had presented several constructions of SLFNs with sufficient hidden neurons. However, in practice, the number of hidden neurons required to achieve a proper generalization performance on novel patterns is much less. And the resulting training error might not approach to zero but can be minimized by solving the following problem:

$$\min_{w_i, b_i, \beta} H(w_1, \dots, w_N, b_1, \dots, b_N) \beta - T^2, \quad (16)$$

Where

$$T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \quad (17)$$

The ELM randomly assigns and fixes the input weights w_i and biases b_i based on some continuous probability distribution function in the case of learning a structured function, only leaving output weights β_i to be adjusted according to:

$$\min_{\beta} H \beta - T^2 \quad (18)$$

The above problem is well established and known as a linear system optimization problem. It is a unique least-squares solution with minimum norm and is given by:

$$\hat{\beta} = HT \quad (19)$$

where H is the Moore-Penrose generalized inverse of the matrix \mathbf{H} . As analyzed by Bartlett (1998) and Huang (2006), the generalization performance of a SLFN tends to be better with smaller magnitude of output weights. From this sense, the solution produced by the ELM in (19) not only achieves the minimum square training error but also the best generalization performance on novel patterns. Huang *et al.* (2004) summarize ELM as the follows:

ELM Algorithm: Given a training set $\mathfrak{S} = \left\{ \left((x_k, t_k) \mid x_k \in \mathbb{R}^m, k = 1, \dots, N \right) \right\}$, an activation function $g(x)$, and the number of hidden neurons \tilde{N} ,

- i. Randomly assign input weights w_i and biases b_i according to some continuous probability density function.
- ii. Calculate the hidden layer output matrix \mathbf{H} .
- iii. Calculate the output weights β_i : $\hat{\beta} = \mathbf{H} \mathbf{T}$.

All the input data are scaled so that they have ranges between -1 to 1 . In general, the SLFN trained by the ELM network starts by randomly choose the input weights which linking the input nodes to the hidden nodes and the hidden neurons' biases, After the input weights and the hidden layer biases are chosen arbitrarily, the SLFNs can be simply considered as a linear system and the output weights which linking the hidden layer to the output layer of the SLFNs can be determined analytically through the generalized inverse operation of the hidden layer output matrices (Wang & Huang, 2005, Huang *et al.* 2006).

4. Results and discussion

This section explains the series of experiments conducted and also presents some preliminary results to compare the effectiveness between the LM training algorithm and ELM training algorithm. The experiment was accomplished by using 48 geometrical features extracted from image data set as input variables to the SLFN-ELM and SLFN-LM network. As mentioned earlier, the SVD method was applied to extract all of the images into a data set that implemented as an input for neural network training and testing processes. The three (3) outputs are classed as lorry, bus or motorcycle for classification purposes. The input variables were taken from 3 sets images of motorcycle, bus and lorry. The data inputs for training and testing consist of 215 samples. For training data set are 96 sets of data and used in training process and the other used in testing process.

	'Lorry'	'Bus'	'Motorcycle'	Total
Training	49	14	33	96
Testing	52	19	48	119
Total	101	33	81	215

Table 1. Description of dataset

Table 1 summarizes the description of the dataset. The SLFN-ELM network was analysed from 1-100 hidden nodes to find the best performance network. A total of 50 trials was conducted for each hidden nodes to find optimal initialization weight. The SLFN-LM network was analysed from 1 to 100 hidden nodes to find the best performance network. However, for the SLFN trained by the LM training algorithm, a total of 10 trials was

conducted since the training algorithm takes a very long time to train SLFNs using back propagation (BP) learning algorithm. The simulation for all networks was conducted in Matlab R2008b using a laptop with Intel Core2Duo 2.4 GHz CPU processor and 4G of RAM. Table 2 tabulates the classification performance of the SLFN-ELM and SLFN-ELM. The comparison is done based on the classification accuracy, training time and optimum structure of each network.

4.1 SLFN-ELM classification performance evaluation

The relationship between the classification performance and number of hidden nodes for SLFN-ELM network is presented in Figure 9. The training and testing results of SLFN network demonstrated that the training phase with ELM algorithm are perform in very fast time to achieve the maximum accuracy after hidden nodes 30. The slope of accuracy against number of hidden nodes rises quickly starting from hidden nodes 6. The accuracy of training with ELM achieves 77.0833% and testing 74.083% at hidden nodes 6. The training and testing accuracy are seemed similar in between hidden nodes 15 to 27. The ELM is able to achieve 100% training accuracy after hidden nodes 77 however the testing accuracy is rather out performed less than 90%.

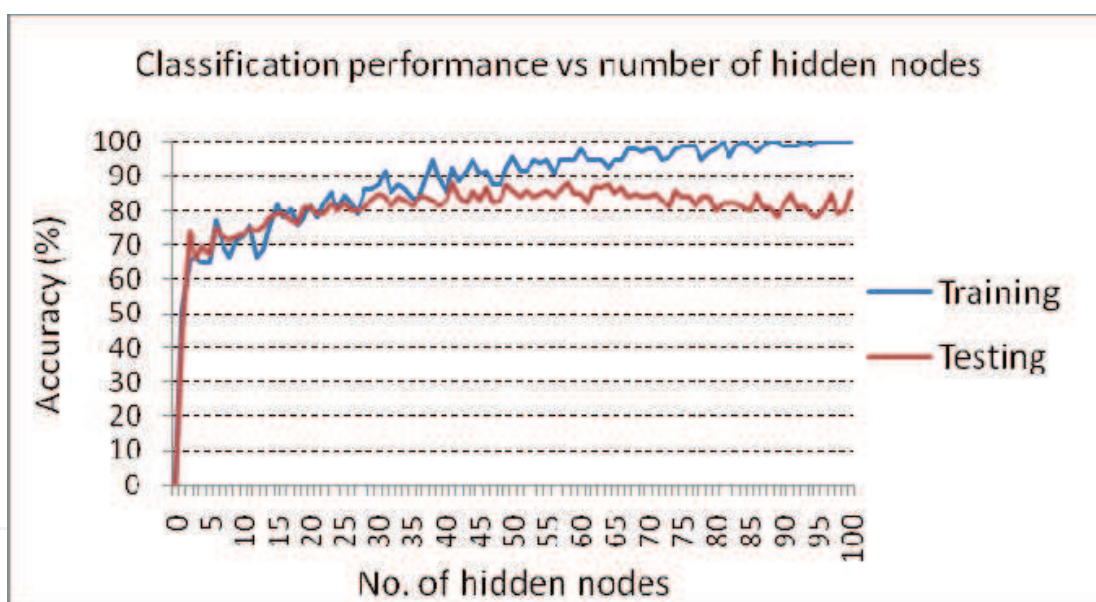


Fig. 9. Performance of the SLFN network trained by Extreme Learning Machine algorithm.

4.2 SLFN-LM classification performance evaluation

The relationship between the classification performance and number of hidden nodes for SLFN-LM network is presented as shown in Figure 10. The training and testing results are plotted in blue and red colour verified that the training phases with LM algorithm has performed timely fast to achieve the maximum accuracy after the second hidden nodes. The slope of accuracy against number of hidden nodes rises quickly starting from the first hidden nodes. The accuracy of training with ELM achieves 77.0833% and testing 75.6303% at first hidden nodes. The training and testing accuracy are seemed similar in between hidden nodes 16, 53 and 85. The LM algorithm is unable to achieve 100% training accuracy but generally it has a slight advantage in testing accuracy where its performance is similar to

ELM. This is showing that testing accuracy for LM algorithm and ELM algorithm having the same performances.

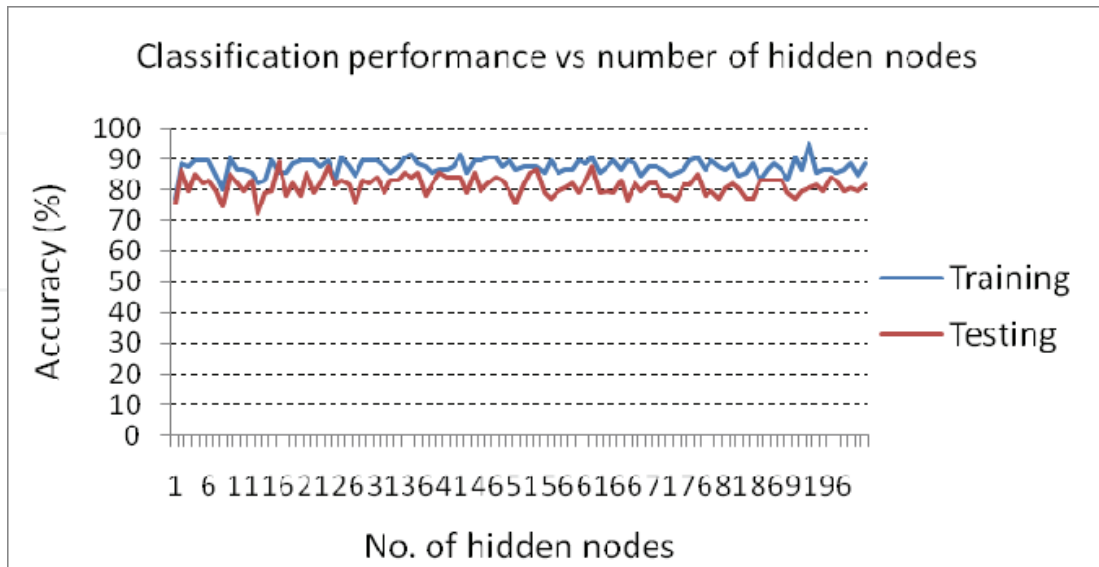


Fig. 10. Performance of the SLFN network trained by Levenberg-Marquardt algorithm

4.3 The best performance of SLFN-ELM and SLFN-LM network

The best classification performance for the SLFN-ELM and the SLFN-LM networks is analysed in section 4.1 and 4.2. The best network for the SLFN-ELM and the SLFN-LM is show in the Table 2. The best network for the SLFN-ELM is at hidden nodes 41 with testing accuracy of 88.253% although the training accuracy for the SLFN-ELM network achieves 100% training accuracy at certain hidden nodes. The best network for SLFN-LM is at hidden nodes 16 with testing accuracy of 89.0756%. These results show that SLFN-LM has possibly a slight difference of a better performance against SLFN-ELM network in term of classification accuracy. The result for training accuracy is better for SLFN-ELM network because the network can achieve 100% accuracy. However the training accuracy is not a significant criterion for classification comparison. This could be due to over-fitting problem, stopping criteria, learning rate, learning epochs and local minima.

Classifier	Accuracy (%)		Training time (s)		Hidden nodes
	Training	Testing	(seconds)	Speedup	
SLFN-ELM	92.7083	88.2353	0.0156	2426.4	41
SLFN-LM	84.375	89.0756	37852	1	16

Table 2. Classification Performance for the SLFN network using the ELM and LM training algorithm

The training time (s) for training SLFN-ELM and SLFN-LM network specify that the SLFN-ELM network is extremely fast and incomparable to the SLFN-LM network. As observed from Table 2, SLFN-ELM classifier can run 2426.4 times faster than SLFN-LM in the case when best classification performances are obtained for both SLFN-ELM and SLFN-LM.

SLFN-ELM network having a tendency to have better classification performance and can be implemented easily in the vehicle recognition system for classification purposes.

4.4 Classification performance for each vehicle using the SLFN-ELM networks

Classification performance for each vehicle using the SLFN-ELM networks is shown in Figure 11. The performance of training and testing in the scale percentage of accuracy is plotted to emphasize the problems of the network in order to recognise each vehicle precisely. From the result, it can be verify that the SLFN-ELM network are able to classify correctly up to 94.74% in percentage of testing accuracy to recognize the bus. Even as the network are only able to classify correctly up to 79.17% for lorries. These results indicate that the SLFN-ELM network is already achieving the convergence despite the fact that one of the testing accuracy of the vehicle achieves the possible maximum accuracy. As observed from Figure 11, the SLFN-ELM can classify a bus better than a lorry because the features of bus are more constant compare to lorry. In this study the type of vehicle car is excluded in order to reduce the complexity of the programming algorithm and to speed up the training and testing time for both classifier network. Furthermore the results of classification performance for vehicles with constant shape such as busses and motorcycles have achieve the accuracy of 94.74% and 94.23% respectively for correct classification in the testing phase.

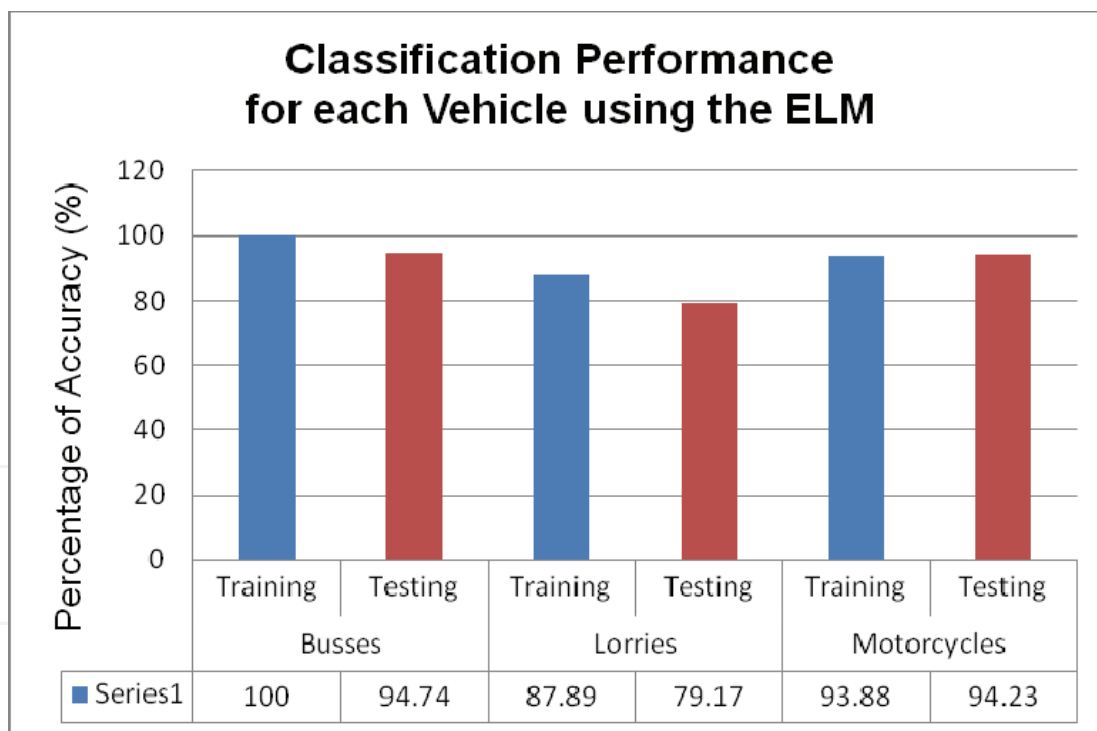


Fig. 11. Classification Performance for each Vehicle using SLFN network trained by ELM training algorithm.

4.5 Classification performance for each vehicle using SLFN-LM networks

Classification performance for each vehicle using SLFN-LM networks is shown in Figure 12. From the result, it can be prove that the SLFN-LM network are also able to classify busses, lorries and motorcycles correctly up to 94.83%, 79.23% and 94.08% respectively in

percentage of testing accuracy. These results also indicate that the SLFN-LM network is already achieving the convergence despite the fact that two of the testing accuracy of the vehicle achieves the possible maximum accuracy. As observed from Figure 12, SLFN-LM also can classify a bus better than a lorry because the features of bus are more constant compare to lorry. In this study the vehicle type car is also excluded in order to reduce the complexity of the programming algorithm and to speed up the training and testing time for both classifier network. Further more the result of classification performance for vehicle with constant shape such as busses and motorcycles have achieve 94.83% and 94.08% respectively for correct classification in testing.

5. Conclusion

In this study, we have evaluated the performance of two main neural network learning algorithm, namely LM and ELM on classification of vehicle type with three classes. The results of this study demonstrate that the ELM needs extremely less training time as compared to conventional LM classifiers. The classification accuracy of ELM is slightly similar to the LM but the ELM is achievable with high accuracy performance. Also, there is significant improvement can be achieved in the testing accuracy for both classifiers by improved the significant features of data input.

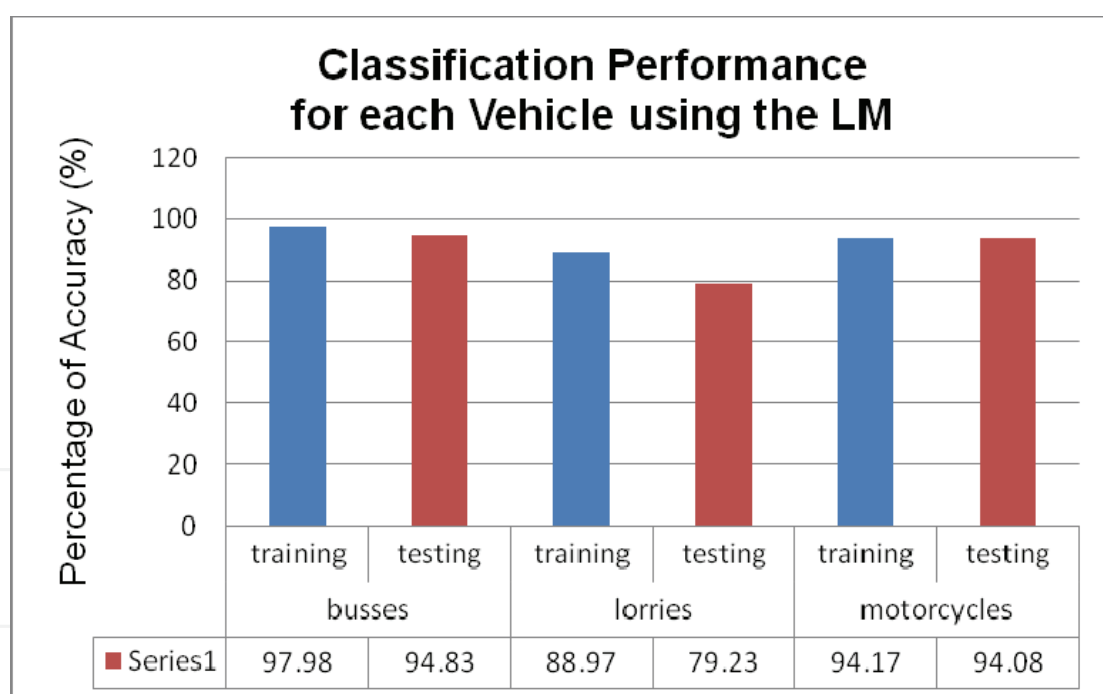


Fig. 12. Classification Performance for each Vehicle using SLFN network trained by the LM training algorithm.

6. References

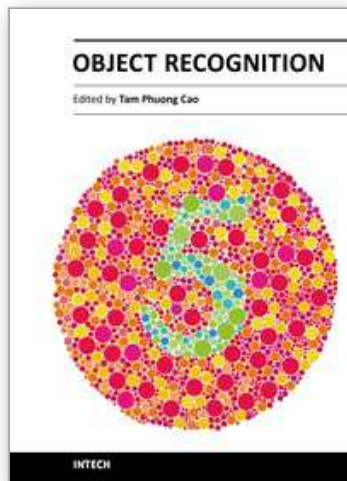
- Barlett, P, L. (1998). The Sample Complexity of Pattern Classification with Neural Networks: the Size of the Weights is more Important than the Size of Network, *IEEE Trans. Inf. Theory*, Vol., 44 No., 2 , (1998), 525 - 536

- Baum, E. B. (1988). On the Capabilities of Multilayer Perceptrons, *Journal of Complexity*, Vol. 4, 193-215, 1988
- Cao, L. (2007) Singular Value Decomposition Applied to Digital Image Processing, Student Trade Show and Project Reports, Arizona State University, Division of Computing Studies, Arizona State University Polytechnic Campus, Mesa, Arizona 85215, (May 2007)
- Huang, G, -B.; Zhu, Q, -Y. & Siew, C, -K. (2006), Extreme Learning Machine: Theory and Applications, *Neurocomputing*, Vol. 70, (May 2006), 489-501
- Hagan, M.T. and Menhaj, M. (1994). "Training Feedback Networks with the Marquardt Algorithm", *IEEE Trans. on Neural Networks*, Vol. 5, Issue 6, No. 6, pp. 989-993. ISSN: 1045-9227
- Huang, G, -B. & Siew, C, -K. (2004). Extreme Learning Machine with Randomly Assigned RBF Kernels, *International Journal of Information Technology*, Vol. 11, No.1, (2004) 16-24
- Huang, G, -B. & Siew, C, -K. (2004). Extreme Learning Machine: RBF Network Case, *Proceeding of the Eighth International Conference on Control, Automation, Robotics and Vision (ICARCV 2004)*, Kunming, China, Dec 2004
- Liang, N, -Y.; Saratchandran, P.; Huang, G, -B. & Sundarajan, N. (2006). Classification of Mental Tasks from EEG Signals Using Extreme Learning Machine, *International Journal of Neural System*, Vol.16, No.1, (2006) 29-38
- Mashor M.Y. (2000). "Performance Comparison Between Back Propagation, RPE And MRPE Algorithms For Training MLP Networks", *International Journal of the Computer, the Internet and Management*, Vol. 8, No.3, 2000
- Mashor, M.Y. (2004). „Performance comparison between HMLP, MLP and RBF networks with application to on-line system identification“, *2004 IEEE Conference on Cybernetics and Intelligent Systems*. Singapore, Dec 2004
- Omar, S.; Saad, Z.; Osman, M.K. and Isa, I. (2010), Improved Classification Performance for Multiple Multilayer Perceptron (MMLP) Network Using Voting Technique, *Proceeding of IEEE, Fourth Asia Modelling Symposium (AMS 2010)*, Sabah, Malaysia, 2010.
- Saad, Z.; Alias, M, F.; Mat Isa, N, A.; Zamli, K, Z. & Sulaiman, S, A. (2007), Application of Image Processing Technique to Determine Normal Sperm, *Proceedings of the International Conference on Robotics, Vision, Information and Signal Processing ROVISIP 2007*, Penang, Malaysia, Nov 2007
- Verma, B. and Ghosh, R. (2002). A Novel Evolutionary Neural Learning Algorithm, *Proceedings of the 2002 Congress on Evolutionary Computation*, pp. 1884-89, Honolulu, Hawaii, USA, 2002.
- Wang, D. & Huang, G, -B. (2005). Protein Sequence Classification Using Extreme Learning Machine, *Proceeding of International Joint Conference on Neural Networks*, Montreal, Canada , Aug 2005
- Yeu, C, -W, T.; Lim, M, -H.; Huang, G, -B.; Agarwal, A. & Ong, Y, -S. (2006). A New Machine Paradigm for Terrain Reconstruction, *IEEE Geoscience and Remote Sensing Letters*, Vol. 3, No.3, (Jul 2006) 382-386

- Zhang, R.; Huang, G, -B.; Sundararajan, N. & Saratchandran, P. (2007). MultiCategory classification Using an Extreme Learning Machine for Microarray Gene Expression Expression Cancer Diagnosis, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, Vol. 4, No.3, (Jul 2007) 485-495
- Zhu, Q, -Y.; Qin, A, K.; Suganthan, P, N. & Huang, G, -B. (2005). Evolutionary Extreme Learning Machine, *The Journal of the Pattern Recognition Society*, Vol. 38, (Mar 2005) 1759-1763

IntechOpen

IntechOpen



Object Recognition

Edited by Dr. Tam Phuong Cao

ISBN 978-953-307-222-7

Hard cover, 350 pages

Publisher InTech

Published online 01, April, 2011

Published in print edition April, 2011

Vision-based object recognition tasks are very familiar in our everyday activities, such as driving our car in the correct lane. We do these tasks effortlessly in real-time. In the last decades, with the advancement of computer technology, researchers and application developers are trying to mimic the human's capability of visually recognising. Such capability will allow machine to free human from boring or dangerous jobs.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Zuraidi Saad, Muhammad Khusairi Osman, Iza Sazanita Isa, Saodah Omar, Sopiaah Ishak, Khairul Azman Ahmad and Rozan Boudville (2011). Vehicle Recognition System Using Singular Value Decomposition and Extreme Learning Machine, Object Recognition, Dr. Tam Phuong Cao (Ed.), ISBN: 978-953-307-222-7, InTech, Available from: <http://www.intechopen.com/books/object-recognition/vehicle-recognition-system-using-singular-value-decomposition-and-extreme-learning-machine>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen