Hybrid Feature Extraction Approach for Handwritten Character Classification Using Feed-forward Neural Network Techniques

Riya Jain M.Tech. Student Sachdeva Institute of Technology, Mathura Gunjan Singh Assistant Professor Raja Balwant Singh Management Technical Campus, Agra Pankaj Sharma Professor Sachdeva Institute of Technology, Mathura

ABSTRACT

Automatic character recognition is one of the most discovered and crucial application areas of pattern recognition field. Due to the increasing demand of machine processing of handwritten characters, this area of pattern recognition is gaining a lot of attention of researchers. Despite of a great deal of efforts done so far in this direction, still a lot is required to get done. In this work, we are focusing our attention on handwritten Hindi characters. Automatic recognition of Handwritten Hindi characters is complex due to the cursive nature and high level of similarity (such as presence of header line, vertical bar, etc.) in the structure of characters. Hybrid feature extraction approach is followed to extract meaningful features from the collected handwritten characters data. A comparative study of performance analysis of selected neural networks models is performed. Results indicate that the Radial basis function network model perform with 99.09% recognition accuracy.

General Terms

Pattern Recognition, Neural Network, Hybrid Feature Extraction.

Keywords

Hindi Character Recognition, Backpropagation Algorithm, Radon Transform, Gabor Filter, Feed-forward Neural Network and Radial Basis Function Network.

1. INTRODUCTION

Automatic character recognition or ACR has been a challenging research area of pattern recognition. Due to the applicability of automatic character recognizers in diverse complex application areas like processing of postal addresses, bank documents, office documents & records, as well as in online dictionaries, digital libraries, automatic translators, language processors, manuscript analysis & processing etc.[1], it has become one of the most explored and prominent application domain of pattern recognition. This area of pattern recognition deals with the machine simulation of human readable characters and aim is to make machine capable to study, analyze, understand and recall characters input to it.

Human readable characters may be machine printed or handwritten, offline or online, cursive or non-cursive etc. Whatever may be the type of the character; machine considers each input character as an image, analyzes it by applying the proposed methods & techniques and finally performs classification, clusterization or mapping as required in the problem statement of the pattern recognition task. Classification of characters is the most investigated problem domain among the various sorts of ACR problems explored till date. Based on the increasing demand and applicability, maximum research has been done for handwritten characters. Due to the varying writing styles, handwritten characters are non-uniform in structure [2]. Characters may also be incomplete, improperly written, combined or overlapped and also same characters, written at different time points, may have different width, orientation and thickness [3].

Proposed work is devoted to the classification and recognition of handwritten Hindi characters. Hindi characters are cursive in nature and have lots of similarities like presence of header line or SHIROREKHA and vertical bar and so on. Also, few characters have almost similar structure like e & O, Ä & /k, etc. These features make recognition of handwritten Hindi characters, a complex problem. Lots of efforts have been done for handwritten Hindi character recognition till date using various pattern recognition techniques such as artificial neural networks [4-8], fuzzy logic [9], support vector machines [10][11], evolutionary techniques [12] etc. Due to these sincere and continuous efforts in the field, research in handwritten Hindi character recognition is getting mature significantly. But as compared to other international official languages such as Chinese, Japanese, Germen, Arabic, etc.; there is still a huge gap between what is required and what has been done. Among various pattern recognition techniques, artificial neural network perform well with significant rate of accuracy. Due to its adaptability [13], Recalling [14] and storing capabilities, ANN is being used broadly and successfully merged with many techniques and procedures applied on handwritten characters. Backpropagation (BP) algorithm is the most commonly used supervised learning algorithm to make a multi-layer feed-forward neural network learn and recall the stored patterns [15]. Although, it suffers with the local minima and slow convergence speed problem, it is the most explored technique yet.

In the present work, we are analyzing the performances of multilayer feed-forward neural network model and Radial basis function network model trained with Backpropagation algorithm for the recognition of handwritten Hindi characters. Networks are trained with six sets of feature patterns. A comparative study of performances and recognition accuracy of selected networks is also done on the basis of the results and conclusions of experiments.

This paper is further organized in five sections. In section 2, feature extraction processes selected for the experiment is stated. In section 3, implementation details of multilayer feed-forward neural network model and radial basis function network used for the experiment are presented. Simulation results, comparative study of performances of the selected neural network models and a complete discussion of the results is presented in section 4. Section 5 considers the conclusion followed by references.

2. FEATURE EXTRACTION

In this work, feature set is created by applying three feature extraction techniques in a sequence. To maintain the reliability and authenticity of the work, 336 samples of handwritten characters are collected by seven different people of different age groups. Each writer was asked to write each of the 48 characters on paper once.

Out of these 336 characters, 288 characters (6 samples of each character) are used as training samples and rest 48 characters (1 sample of each character) are considered as the sample for testing. Training pattern sets are termed as F1, F2, F3, F4, F5 and F6; while test pattern set is termed as F7. Collected samples are scanned to convert handwritten data into image pattern and make them suitable to be used for the selected pattern classification task of Hindi characters using neural networks. After scanning, a sequence of preprocessing operations is applied on scanned images for further processing.

As collected images may be of different sizes so it is essential to resize scanned images to a predefined size. Thus, first scanned images are resized to the predefined size of 30x30. After resizing, RGB images are converted to gray scale form and then to the binary form using global thresholding method. Results of applied preprocessing operations are shown in fig. 1.





Total seven feature vectors are created, out of which six are used to train and last one is used to test the selected multilayer feed-forward networks. Each feature vector comprises one sample of each character. To create these vectors, three feature extraction techniques are applied sequentially on binary images of characters. First Gabor filter technique is applied to filter the binary image, to remove noise elements (if any) and enhance the character image for further processing. Originally introduced by Dennis Gabor in 1946, Gabor filter gives the highest response at edges and in cases where the texture changes. It makes Gabor filter appropriate to be applied on text images. A Gabor filter can be one-dimensional, two-dimensional [16] [17] or three dimensional [18]. One-dimensional Gabor filter is defined as the multiplication of a cosine (even) /sine (odd) wave with Gaussian window expressed as equation (1) and (2) respectively as [19]:

$$g_e(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}\cos(2\pi\omega_0 x)}$$
(1)

$$g_{o}(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^{2}}{2\sigma^{2}}\sin(2\pi\sigma_{0}x)}$$
(2)

where ω_0 : frequency in which the filter gives the highest response (also called the center frequency), σ : spread of the Gaussian window.

One-dimensional Gabor filter is applied for waves. For images, two-dimensional Gabor filter is used. Twodimensional Gabor filter is expressed respectively as equations (3) and (4) as:

$$g_{e}(x, y) = \frac{1}{\sqrt{2\pi}\sigma_{x}\sigma_{y}}e^{-\frac{1}{2}(\frac{x^{2}}{\sigma_{x}} + \frac{y^{2}}{\sigma_{y}})\cos(2\pi\omega_{x0}x + 2\pi\omega_{y0}y)}$$
(3)

$$g_{o}(x,y) = \frac{1}{\sqrt{2\pi\sigma_{x}\sigma_{y}}} e^{-\frac{1}{2}(\frac{x^{2}}{\sigma_{x}} + \frac{y^{2}}{\sigma_{y}})\sin(2\pi\sigma_{x0}x + 2\pi\sigma_{y0}y)}$$
(4)

where $(\omega_{xo}, \omega_{yo})$: center frequency and (σ_x, σ_y) : spread of the Gaussian window.

Gabor filter is applied to the given set of character images with the values of gamma (aspect ratio) = 0.3, psi (phase) = 0, theta (orientation) = 90, bw (bandwidth) = 3.6, lambda (wavelength) = 4.6 and pi = 180.

After filtration, Radon transformation is applied on filtered character images. Radon transform has been successfully used in many line detection applications as it transforms a two-dimensional image with lines into a domain of probable parameters. Each line in the image gives a peak which is positioned at the corresponding line parameters [20]. Radon transform of an image f(x,y) for a given set of angles can be computed by calculating the projections of the image along the given angles [21], where projection of a two-dimensional function f(x,y) is a set of line integrals. A point in the projection $g(\rho_j, \theta_k)$ is the ray-sum along $x \cos \theta_k + y \sin \theta_k = \rho_j$

The Radon transform calculates the line integrals obtained from multiple sources along parallel paths (beams) in a certain direction. Parallel beam projections of the image are taken from different angles by rotating the source around the center of the image as shown in fig. 2.



Fig.2. Projection at a particular rotation angle

Expression for Radon transformation to calculate projections along any angle, say θ , is [22]:

$$R_{\Theta}(x') = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \Theta + y \sin \Theta - x') dx dy$$
⁽⁵⁾

Discrete Radon transform of an image f (x, y) at coordinates (ρ, θ) can be expressed as [23]:

$$g(\rho,\theta) = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} f(x,y) \delta(x\cos\theta - y\sin\theta - \rho)$$
⁽⁶⁾

Finally, Eigen value computation method is applied on transformed character images. A scalar λ is called the Eigenvalue of NxN size matrix, say A, if there exists a non-trivial solution x of Ax- λ x [24]. In image processing, the Eigen value computation method is used to reduce the size of image matrix. An image I can be reduced by using the following transformation [25]:

$$\mathbf{X} = \mathbf{A}^{\mathrm{T}} \mathbf{x} \mathbf{I} \mathbf{x} \mathbf{A} \tag{7}$$

Here, columns of the matrix A represent the normalized Eigen vectors of the image I and diagonal elements are the Eigen values of the image I. Using this method all low coefficients of an NxN size image are vanished and only high coefficients are remained. Thus, the image is normalized without any loss of information. The reduction in the size of image matrix results in reduced transmission time and compact image storage.

Obtained character images produces vectors with high dimensionality, which may cause slow computation speed during training of the network and may also degrade the performance of the network. To overcome the problem and to produce vectors with low dimensionality, Singular Vector Decomposition (SVD) is applied. SVD is widely used in many image processing applications such as image restoration, image compression and object recognition [25]. SVD works by taking a high dimensional, highly variable set of data points and reduce it to a lower dimensional space. Reduction is performed by considering the most relevant and significant data points and ignoring the remaining points for the classification process. Using SVD method, an image matrix A of size m x n is factored into three matrices expressed as [26]:

$$\mathbf{A}_{mn} = \boldsymbol{U}_{mm} \boldsymbol{S}_{mn} \boldsymbol{V}_{nn}^{T} \tag{8}$$

where U: an orthogonal matrix, S: a diagonal matrix and V : transpose of orthogonal matrix

Obtained singular values are unique. By considering only the leading singular points of the image, an approximation of the image is obtained. Results of applying selected feature extraction methods are shown below in fig.



Fig.3. Results of feature extraction Methods: (a) Gabor transform, (b) Radon transform, (c) Eigen value method, and (d) Singular vector decomposition

3. IMPLEMENTATION OF NEURAL NETWORK MODELS

An artificial neural network (ANN) is a computational model which is designed to perform complex pattern recognition tasks. Therefore, a neural network can be termed as a computing architecture designed to resemble the learning and storing capability of human brain for performing the task of pattern recognition [27]. Neural network learns from a given training feature pattern vector (set). Trained network performs the required character identification based on the given learning algorithm. On inputting an unknown feature pattern, the network looks for the similar properties in the target character image to be identified.

In this simulation, we created three feed-forward neural network models trained with backpropagation learning algorithm. Backpropagation is a supervised learning algorithm. It is the base of all delta learning rule. There is a predefined desired target output (t) with each input pattern prepared for the network. Actual output (y) of the network is compared with target (t) and error (E) is computed. Error is propagated backward from the output layer towards the input layer and weights in the hidden layer/s and output layer are updated accordingly [28] Equations of weight updation in hidden layer/s and output layer are given as [29]:

$$v_{kj} = v_{kj} + c\lambda(t_k - z_k)z_k(1 - z_k)y_i$$
(9)

and

$$w_{ji} = w_{ji} + c\lambda^2 y_i (1 - y_i) x_i \left(\sum_{k=1}^{K} (t_k - z_k) z_k (1 - z_k) v_{kj} \right)$$
(10)

where W_{ii} : weights connecting input layer ith neuron to the

 j^{th} neuron of the hidden layer, \mathcal{V}_{kj} : weights connecting hidden layer j^{th} to the k^{th} neuron in the hidden layer, λ : parameter used to control the gradient of the function, t_k : output of the k^{th} target vector and y_i : output of the net output of the hidden layer neuron.

Error term is represented as [28]:

$$E = \frac{1}{2} \sum_{k} [t_k - y_k]^2$$
(11)

Out of the three networks created to perform the proposed character classification task, first two networks are selected as multilayer feed-forward neural networks trained with optimal backpropagation learning rule and third as Radial basis function neural network. In a multilayer feedforward neural network, input vector is provided to the input layer & passes to the output layer through one or more hidden layers and the output is calculated [27]. Two different arrangements, called method1 and method2 respectively, are designated for multilayer Feed-forward networks. Chosen methods differ by the size of the target vector. In the first method, target vector is taken as a matrix of size 48x48; while in the second method, it is considered as a vector of size 1x48. Parameters used to create the three networks are shown as below in table 1.

Table 1. Parameters used to create two multilayer feed-forward neural networks

	Value			
Parameter	Network with Method 1	Network with Method 2		
Number of hidden layer	1	1		
Number of nodes in hidden layer	17	17		
Size of target vector	48x48	1x48		
Transfer function	Log-sigmoid	Log-sigmoid		
Training function	Levenberg- Marquardt	Levenberg- Marquardt		
Maximum number of epochs	1000	1000		
Performance function	Mean squared error	Mean squared error		
Error goal	0.00001	0.00001		
Backpropagation learning rate	0.1	0.1		
Adaption rate	1.0	1.0		

Another neural network selected for the experiment in this work is Radial basis function network. It has a simpler structure and faster training process than a multilayer Perceptron model. It was devised to solve the problem of exact interpolation of a set of data points in a multidimensional space [30], thus it is a universal approximator and is widely used in the areas of interpolation and approximation theory. In RBF, there exists only a single hidden layer in the structure of RBFN as presented in fig. 4. Hidden layer accomplishes the nonlinear transformation of input pattern data and output layer performs as a linear combiner which maps this nonlinearity into a new space, thus hidden layer is nonlinear while output layer is linear in nature.



Fig.4: Radial Basis Function Network

The number of units in the hidden layer is less than the number of training samples and each unit in the hidden layer implements a radial basis function such as Gaussian function, Multi-quadric function, Inverse Multi-quadric function, Thin plate spline function, cubic, logistic function, etc. These radial basis functions are special classes of functions and the value of the function increases or decreases monotonically with distance for the center point. In pattern classification problem, Gaussian function is widely used to compute the activation value of the units of the hidden layer by computing the Euclidean norm (distance) between the input vector and the center of that unit expressed as [31]:

$$\phi_i(X) = e^{\frac{\|x - c_i\|^2}{2\sigma_i^2}}$$
(12)

 $\begin{array}{l} \mbox{for $i=1,2,3,\ldots,$number of nodes in hidden layer} \\ \mbox{where $x: J_1$ dimensional input vector, σ_j : width of the neuron, μ_j : center of the i^{th} node and $\|.\|$: denotes Euclidean norm (distance) } \end{array}$

Therefore, output of the network can be expressed as [31]:

$$y_i(x) = \sum_{k=1}^{N} w_{ki} \phi(||x - c_k||)$$
(13)
for i=1,2,...., number of nodes in output layer

where $y_i(x)$: ith output of the network, w_{ki} : weight from the kth unit in the hidden layer to the ith unit in the output layer, x: input pattern, and c_k: center of the kth hidden unit,

Parameters used to create the network are shown in table 2.

Table 2.	Parameters	used to	create	Radial	basis	function	network

Parameter	Value
Spread of Radial basis function	1.0
Performance function	Mean squared error
Transfer function in layer 1	Radial basis transfer function
Transfer function in layer 2	Linear transfer function
Backpropagation learning rate	0.1
MSE	0.000315657

4. RESULTS AND DISCUSSION

In the proposed simulation, performance of all neural network models is analyzed for the created feature vector. The results from the simulations are considered from both selected feed-forward neural network and Radial basis function network models. Performance of these neural network models for their recognition accuracy is presented in table 1. The regression between simulated output of the network performance and expected output for the input patterns is obtained to exhibit the performance analysis.

Table 3: Regression value of selected neural network models for training and testing pattern set

		Regression value R			
Feature Vector		Feed-forward network with method 1	Feed-forward network with method 2	Radial basis function network	
aini g	F1	0.4379	0.6574	1	
Tri n	F2	0.3809	0.7158	0.9887	

	F3	0.454	0.5937	1
	F4	0.4764	0.6415	1
	F5	0.4313	0.6383	1
	F6	0.4309	0.7008	0.9568
Averag	ge R			
for tra	ining	.4352	.6579	.9909
Test				
-ing	F7	0.3647	0.5406	0.9358

A comparative analysis of selected neural network models for training and testing feature patterns is presented in fig. 5 and fig. 6 respectively. These figures show the comparison of the performance of both feed-forward networks and radial basis function network models.



Fig.5. Comparison of regression values of training patterns of selected neural network models



Fig.6. Comparison of regression values of average of all training patterns and testing patterns of selected neural network models

The above mentioned results show that the Radial basis function network performs better for all the feature vectors than both feed-forward neural network models. The Radial basis function network exhibits 100% recognition accuracy for four training pattern vectors. For rest two feature vectors, the network shows above than 95% recognition accuracy. Thus, RBF demonstrate good approximation and generalization characteristics as well as capabilities for the selected feature vectors. Thus the network is generalized well and shows reasonable approximation. Results also show that the feed-forward network with method 2 gives better results for all the feature vectors as compare to the feed-forward network with method 1. Results indicate that neural network with method 1 gives 38%-47% recognition accuracy for all training pattern vectors; while network with method 2 gives 59% - 71% recognition accuracy for all training vectors. Therefore, the network with method 2 shows more variability in recognition rate as compared to the first network. Results indicate that network with method 2 show highest recognition accuracy for feature set F2 and network with method 1 show the lowest recognition accuracy for this feature vector.

5. CONCLUSION

In this paper, we analyzed the performance of three selected feed-forward neural network models for the recognition of handwritten Hindi characters. Seven (7) feature vectors are created by applying three well-known feature extraction methods – Gabor transform, Radon transform and Eigen value method in a series. Finally, Singular value decomposition (SVD) method is applied on resultant feature vectors to create six (6) training patterns and one (1) testing pattern. Simulated results of the performance evaluation are presented and discussed. On the basis of the simulated performance evaluation, following observations has been drawn:

- (i) Simulated results are specifying that the Radial basis function network model shows 99.09% recognition accuracy for training pattern and 93.58% recognition accuracy for testing pattern. Therefore, the network is exhibiting good generalization and approximation behavior for the training and test pattern vectors.
- (ii) Results indicate that feed-forward neural network model with method 1 gives 43.52 % and 36.47% recognition accuracy for the created training and testing pattern vectors respectively.
- (iii) Results also show that feed-forward neural network model with method 2 gives 65.79 % and 54.06% recognition accuracy for the created training and testing pattern vectors respectively.
- (iv) Results show that feed-forward network with method 2 performed better than network with method 1. Therefore, network with target vector gave better result than network with target matrix for the created feature pattern vectors.
- (v) Modular neural network performed very well to recognize handwritten characters. The future work is devoted to implement modular neural networks to improve the performance and increase the recognition accuracy.

REFERENCES

- Govindan V.K. and Shivprasad A.P. 1990. Character Recognition: A Review. Pattern Recognition, vol. 23, no. 7, pp. 671-683.
- [2] Plamondon R. and Srihari S.N. 2000 Online and Off-line Handwriting Recognition : A Comprehensive Survey. IEEE Transcations on Pattern Analysis and Machine Intelligence, vol. 22, issue 1, pg. 63-84.
- [3] Bunke H. and Wang P.S.P. 1997. Hand Book of Character Recognition and Document Image Analysis. World Scientific.
- [4] Singh G. and Lehri S. 2012. Recognition of Handwritten Hindi Characters using Backpropagation Neural Network

International Journal of Computer Science and Information Technologies, vol. 3(4), pg. 4892-4895.

- [5] Sayyad S. S., Jadhav A., Jadhav M. Miraje S., Bele P. and Pandhare A. 2013. Devnagiri Character Recognition Using Neural Networks, vol. 3, issue 1pg. 476-480.
- [6] Singha D, Sainib J.P. and Chauhan D.S. 2014. Analysis of Handwritten Hindi Character Recognition Using advanced Feature Extraction Technique and Backpropagation Neural Network. International Journal of Computer Applications, vol. 97, no. 22, pg. 7-14.
- [7] Singh G., Kumar S. and Singh M.P. 2017. Performance Evaluation of Feed-Forward Neural Network Models for Handwritten Hindi Characters with Different Feature Extraction Methods. International Journal of artificial Life Research, vol. 7, issue 2, pg. 38-57.
- [8] Dogra S. and Sehgal A. 2017. A Framework for Segmentation and Recognition of Hindi Letters. International Journal for Scientific Research & Development, vol. 5, issue 4, pg. 1095-1101.
- [9] Hanmandlu M., Ramana Murthy O.V. and Madasu V.K. 2007. Fuzzy Model based Recognition of Handwritten Hindi characters. 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications, IEEE, 3-5 Dec.
- [10] Gaur A. and Yadav S. 2015. Handwritten Hindi Character Recognition Using k-means Clustering and SVM. 4th International Symposium of Emerging Trends and Technologies in Libraries and Information Services, IEEE, pg. 65-70.
- [11] Kakde P.M. and Gulhane S.M. 2016. A Comparative Analysis of Particle Swarm Optimization and Support Vector Machines for Devnagri Character Recognition: An Android Application. 7th International Conference on Communication, Computing and Virtualization, Elsevier, vol. 79, pg. 339-343.
- [12] Kumar S, Singh M.P., Goel R. and Lavania R. 2013. Hybrid Evolutionary Techniques in Feed forward Neural Network with Distributed Error for Classification of Handwritten Hindi 'SWARS'. Connection Science, vol. 25, no. 4, pg. 197-215.
- [13] Barto A.G., Sutton R.S and Anderson C. 1984. Neuronlike Adaptive Elements that can Solve Difficult Learning Control Problems. IEEE Transaction on Systems, Man and Cybernetics, Vol. 13, pg. 834-846.
- [14] Kumar S. and Singh M.P. 2010. Pattern Recalling Analysis of English Alphabets using Hopfield Model of Feedback Neural Network with Evolutionary Searching. International Journal of Business Information Systems, vol. 6, no. 2, pg. 200-218.
- [15] Yahya H.Z., Lakmal D., Seneviratne K.A. 2005. Stability Analysis of a Three-term Backpropagation Algorithm. Neural Networks, vol. 18, no. 10, pg. 1341-1347.

- [16] Duagman J. 1980. Two-dimensional Analysis of Cortical Receptive Field Profiles. Vision Research, vol. 20, pg. 846–856.
- [17] Daugman J. 1985. Uncertainty Relation for Resolution in Space, Spatial Frequency and Orientation Optimized by Two-dimensional Visual Cortical Filters. Journal of the Optical Society of America-A, vol 2, no. 7, pg. 1160– 1169.
- [18] Heeger D. 1987. Model for the Extraction of Image Flow. Journal of the Optical Society of America-A, vol. 2, no. 2, pg. 1455–1471.
- [19] Gabor D. 1946. Theory of Communication. Journal of the Institute of Electrical Engineers, vol. 93, pg. 429-457.
- [20] Hoilund C. 2007. The Radon Transform. Aalborg University, VGIS.
- [21] Acharya T. and Ray A. K. 2005. Image Processing -Principles and Applications. Wiley Publications.
- [22] Kupce E. and Freeman R. 2004. The Radon Transform: A New Scheme for Fast Multidimensional NMR. Concepts in Magnetic Resonance, Wiley Periodicals, vol. 22, pp. 4-11.
- [23] Miciak M. 2010. Radon Transformation and Principal Component Analysis Method Applied in Postal Address Recognition Task. International Journal of Computer Science and Applications, vol. 7, no. 3, pp. 33-44.
- [24] Gaidhane V. H., Hote Y. V. and Singh V. 2011. A New Approach for Estimation of Eigenvalues of Images. International Journal of Computer Applications, vol. 26, no. 9, pg. 1-6.
- [25] Shridhar S. 2016. Digital Image Processing. Oxford University Press, Second Edition..
- [26] Sadek R.A. 2012. SVD Based Image Processing Applications: State of The Art, Contributions and Research Challenges. International Journal of Advanced Computer Science and Applications, vol. 3, no. 7, pg. 26-34.
- [27] Yagnanarayana B. 2004. Artificial Intelligence. Prentice Hall Publication, Ninth Edition.
- [28] Sivanandam S.N., Sumathi S. and Deepa S.N. 2005. Introduction to Neural Networks Using MATLAB 6.0. Tata McGraw-Hill Publication, Third Edition.
- [29] Rajasekaran S. and Pai G.A.V. 2012. Neural Networks, Fuzzy Logic and Genetic Algorithms - Synthesis and Applications. PHI Publication, Sixteenth Edition.
- [30] Powell M.J.D. 1987. Radial Basis Functions for Multivariate Interpolation: A Review. In Algorithms for the Approximation of Functions and Data, J.C. Mason and M.G. Cox, eds., Clarendon Press, pp. 143-167.
- [31] Diao Y. and Passino K.M. 2002. Adaptive Neural / Fuzzy Control for Interpolated Nonlinear Systems. IEEE Transactions on Fuzzy Systems, vol. 10, no. 5, pg. 583-595.