Hypothetical Pattern Recognition Design Using Multi-Layer Perceptorn Neural Network For Supervised Learning

Md. Abdullah-al-mamun, Mustak Ahmed

Abstract: Humans are capable to identifying diverse shape in the different pattern in the real world as effortless fashion due to their intelligence is grow since born with facing several learning process. Same way we can prepared an machine using human like brain (called, Artificial Neural Network) that can be recognize different pattern from the real world object. Although the various techniques is exists to implementation the pattern recognition but recently the artificial neural network approaches have been giving the significant attention. Because, the approached of artificial neural network is like a human brain that is learn from different observation and give a decision the previously learning rule. Over the 50 years research, now a day's pattern recognition for machine learning using artificial neural network got a significant achievement. For this reason many real world problem can be solve by modeling the pattern recognition process. The objective of this paper is to present the theoretical concept for pattern recognition design using Multi-Layer Perceptorn neural network(in the algorithm of artificial Intelligence) as the best possible way of utilizing available resources to make a decision that can be a human like performance.

Index Terms: Pattern Recognition, Multi-Layer Perceptron, MLP, Artificial Neural Network, ANN, Backpropagation, Supervised learning

1 INTRODUCTION

Since our childhood, we have been seen different object in different patter around the world like flowers, animals, toys, different character and so on. So children can recognize a simple digit or letter and different type of complex character or handwritten character or partially occurred character can be recognized by the young man. We got the ability to this pattern recognition process by the learning from different observation. In the same way a machine can be intelligence by different learning process to recognize the pattern. Pattern recognition is the study of how machines can observe the environment, learn to distinguish patterns of interest from their background, and make sound and reasonable decisions about the categories of the patterns [11]. Since from 1960, about 50 years research we have reached remains an elusive goal to design a pattern recognition machine. Here, our goal is to introduce pattern recognition using artificial neural network (like-human-brain) as the best possible way of utilizing available sensors, processors, and domain knowledge that in some way could emulate human performance [12].

2 PATTERN RECOGNITION

Pattern recognition is a branch of machine learning that focuses on the recognition of patterns and regularities in data, although it is in some cases considered to be nearly synonymous with machine learning [13]. So it is the important parts of Artificial Intelligence engineering and scientific disciplines that can be provide the elucidation of different fields such as Finance, Hospitals, Biology, Medicine, Robotics, Transportation, and Human-Computer Interaction and so on. So the term pattern recognition encompasses a wide range of information processing problems of great practical significance [11].

But what is the pattern? Generally pattern is an abstract object that describing the measurement of physical object. Watanabe defines a pattern "as opposite of a chaos; it is an entity, vaguely defined, that could be given a name" [12]. So pattern is p-dimensional data vector for input feature of an object xi=(x1,x2,…xp)T (T denotes vector transpose). Thus the features are the variables specified by the investigator and thought to be important for classification [16]. We can illustrate of a pattern is as an image data, optical character data, fingerprint image data, speech signal data and these types of things. So the Pattern recognition studies the operation and design of systems to recognize patterns in data. It encloses sub disciplines like discriminate analysis, feature extraction, error estimation, cluster analysis (together sometimes called statistical pattern recognition), grammatical inference and parsing (sometimes called syntactical pattern recognition) [14]. Interest in the area of pattern recognition has been renewed recently due to emerging applications which are not only challenging but also computationally more demanding [11]. The application area of the pattern recognition is involves with data mining, classification text into several categories from a document, personal identification based on physical attributes like face, fingerprints or voice, financial forecasting, area of image recognition such as optical character recognition, handwriting image detection and so on. Picard has identified a novel application of pattern recognition, called affective computing which will give a computer the ability to recognize and express emotions, to respond intelligently to human emotion, and to employ mechanisms of emotion that contribute to rational decision making [15].

3 ARTIFICIAL NEURAL NETWORK

Modeling systems and functions using neural network mechanisms is a relatively new and developing science in computer technologies [21]. The principle of the structured of artificial neural network are the way of neurons interact and function in the human brain. In the human brain contains approximately ten thousand million (1010) neurons and each of those neurons is connected to about ten thousand (104) others [16]. Neurons in the brain communicate with one another across special electrochemical links known as synapses and each synaptic connection of 1015, which gives

- *Md. Abdullah-al-mamun, Mustak Ahmed*
- *Department of Computer Science and Engineering,*
- *Rajshahi University of Engineering and Technology, Bangladesh.*

the brain its power in complex spatio-graphical computation [21]. Generally, human brain operates as a parallel manner that can be recognition, reasoning and reaction. All these seemingly sophisticated undertakings are now understood to be attributed to aggregations of very simple algorithms of pattern storage and retrieval [21]. The main characteristics of neural networks are that they have the ability to learn complex nonlinear input-output relationships, use sequential training procedures, and adapt themselves to the data [19]. Neural networks also provide nonlinear algorithms for feature extraction (using hidden layers) and classification (e.g., multilayer perceptrons) that can be mapped on neural network architectures for efficient implementation. Ripley [9] and Anderson et al. [8] also discuss this relationship between neural networks and pattern recognition. Anderson et al. point out that neural networks are statistics for amateurs. Most NNs conceal the statistics from the user". Not only neural network verses pattern recognition similarities but also neural networks provides the approaches for feature extraction and classification process that are possible to provide a solution for the data patter whose are linearly non separable. The increasing popularity of neural network models to solve pattern recognition problems has been primarily due to their seemingly low dependence on domain-specific knowledge (relative to model-based and rule-based approaches) and due to the availability of efficient learning algorithms for practitioners to use. To implement the pattern recognition process through the neural network the most commonly used feed-forward network algorithm is Multi-Layer Perceptron, Radial-Basis Function (RBF) and another algorithm for data clustering or feature mapping algorithm is Self-Organizing Map(SOM) or Kohonen network, Hopfield network. Here we want to present Multi-layer Perceptron for the supervised learning.

4 SUPERVISED LEARNING

Pattern recognition is a branch of machine learning that focuses on the recognition of patterns and regularities in data For a give pattern its learning or classification process can be involve the two processes: (1) Supervised learning and (2) Unsupervised learning. Research progress in supervised classification in the past several decades has produced a rich set of classifier technologies. Algorithms for supervised classification, where one infers a decision boundary from a set of training examples, are at the core of this capability [20]. In the supervised learning it is assumes that training data set has been provided and the instance of the data set are properly labeled to produce the correct output. A learning procedure then generates a model that attempts to meet two sometimes conflicting objectives: Perform as well as possible on the training data, and generalize as well as possible to new data (usually, this means being as simple as possible, for some technical definition of "simple", in accordance with Occam's Razor, discussed below)[16]. Generally, the problem of supervised pattern recognition can be describe is as: Consider an ground truth function $g: X \rightarrow Y$ that's maps into input instances $x \in X$ to output labels$y \in Y$ with training data $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ is produce a function $h: X \rightarrow Y$. According to the decision theory, a loss function is assigns a specific value of "loss" that's result is produced an incorrect label. Practically, the distribution of *X* and the ground truth function $g: X \rightarrow Y$ any one can't be known exactly but can be compute the large number of input sample *X* that is collecting by the experimentally and the correct output value *Y*. Our goal is to minimize the error rate of the learning procedure from the input set *X*.X To estimate the probability of output label for a given input instance; the estimate function for a probabilistic pattern recognition,

$$p(label|x, \theta) = f(x; \theta) \quad (1)$$

Where, *x* is the input instances for the feature vector; the function *f* is parameterized by some parametersθ. Using the Bayes' rule, the prior probability $p(label|x, \theta)$ for the input instances is as following,

$$p(label|x, \theta) = \frac{p(x|label, \theta)p(label|\theta)}{\sum_{L \in all\ labels} p(x|L)p(L|\theta)} \quad (2)$$

If the input instance labels are continuously distributed then integration is involves rather than summation for the denominator,

$$p(label|x, \theta) = \frac{p(x|label, \theta)p(label|\theta)}{\int_{L \in all\ labels} p(x|L)p(L|\theta)dL} \quad (3)$$

The value of $\theta$is learned using maximum a posteriori (MAP) estimation. To decreases the error rate of θ, we should be concern the two factors, one is perform the training data with minimum error-rate and design the simplest possible model.

5 PATTERN RECOGNITION DESIGN

In general, pattern recognition process involves preprocessing the data object, data acquisition, data representation like matrix mapping and at last decision making. The decision making process of pattern recognition is associated with classification and learning task, where the classes are defined by the system designer and learned process is based on the similarity of patterns.

5.1 Data Acquisition

Data acquisition for the pattern recognition is like as a real manufacturing process. In pattern recognition process its required large numbers of input instance to enter the recognition process. So to collect data it needs to be a proper preprocessing like noiseless data, no redundant data, optimized window size and same types of matter in a way that cost is optimized. Any image filtering algorithm can be used in here as the required. After collection the data it needs to be a proper representation to feed the neural network as an input instance to generate the series of output according to the desire bit representation. Image pixel to matrix mapping is most commonly used for the data representation. Consider the sampling object is mapped 20x25 binary matrix grid then it would contains 500 elements that will be used for input in the neural network.

5.2 Multi-Layer Perceptron

Multi-Layer Perceptron(MLP) is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate outputs [17]. It consist of two or more layers where each layer contains several numbers of nodes and the node of one layer is fully connected to the next layer all nodes as a direct graph. In the multilayer perceptron, starting layer is an input layer, end layer is output layer and between the input and output layer one or more hidden layers can be associated.

Expect the input layer, a non-linear activation function is associated with each node of all layer in the neural network. Each node of a layer connects with a certain weight Wij where it indicated the ith node to jth node weight.MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable [17].

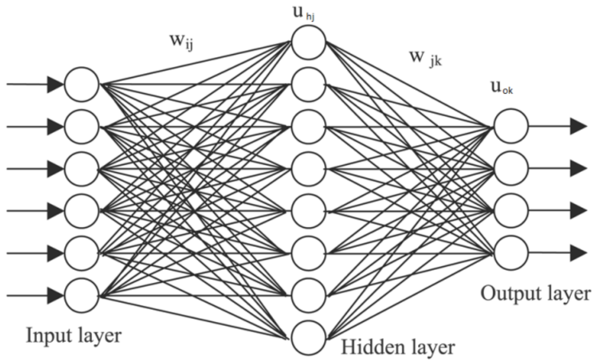

Figure-1: *Multi-layer Neural Network*

5.3 Activation Function

Generally, the activation function of a node defines the output of that node from the given an input or input set. For a multilayer perceptron, the activation function for all neurons maps with the weighted input to produced output of each neuron. Consider the general form of the activation function is as $y_i(x) = \emptyset(\omega'_{ix} + b_i) = \emptyset(z_i)$ (4)

Here, y$_i$ is the output of the i'th node (neuron) and v$_i$ is the weighted sum of the input synapses. There are a variety of non-linear activation functions that may be used is as following,

1. Heaviside (or step) function:

$$y_i(x) = \begin{cases} 0, & z_i < 0 \\ 1, & z_i \geq 0 \end{cases} \quad (5)$$

Sometimes used in this threshold units where output is binary.

2. Sigmoid (or logistic regression) function:

$$y_i(x) = \tanh(x) = (1 + e^{z_i})^{-1} = \frac{1}{(1+e^{z_i})} \quad (6)$$

The range of the logistic function is 0≤ $y_i(x)$ ≤1.

3. Softmax (generalised logistic) function:

$$y_i(x) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \quad (7)$$

The range of the generalised logistic function is 0≤ $y_i(x)$ ≤1

4. Hyperbolic tan (or tanh) function:

$$y_i(x) = \frac{e^{z_i} - e^{-z_i}}{e^{z_i} + e^{-z_i}} \quad (8)$$

The range of hyperbolic tangent is -1 ≤ tanh(x) ≤ +1

The curve of the different activation function is as following,

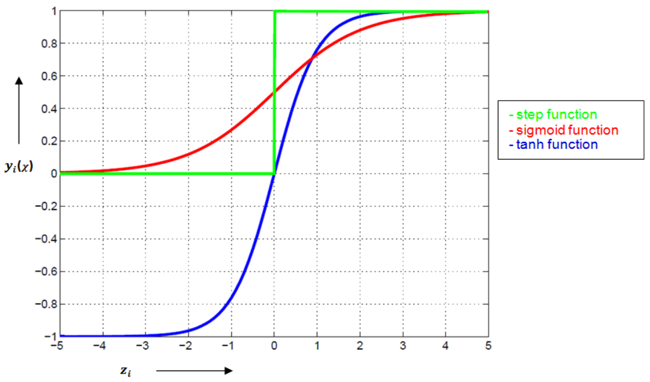

Figure-2: *Comparison amoung the activation funciton.*

5.4 Learning

The learning process involves updating network architecture and connection weights so that a network can efficiently perform a specific classification/clustering task [19]. A *class* of functions F training process means find $f^* \in F$ using a set of *observations* which solves the task in some *optimal* sense. In Multi-Layer perceptorn to implement the learning process error backpropagation learning rule are most uses as it is the supervised learning method.

5.5 Backpropagation Learning rule

Back-propagation or Error Back-propagation is common method for training in artificial neural networks. It can be considered to be a supervised learning method. Also called generalized delta algorithm because it extends the means to train a one layer perceptron (delta rule), it is based on minimizing the difference between the desired output and the real output, through descent error gradient method [10]. In back-propagation learning rule, number of neurons are connected with each other according to the network architecture, an activation function is associated with each neurons and the learning law that is used to for adjusting weights. In this back-propagation learning rule first phase, training input data pattern is feed to the network input layer. The network then propagates the input pattern from layer to layer until the output pattern is generated by the output layer. If this pattern is different from the desired output, an error is calculated and then propagated backwards through the network from the output layer to the input layer [10]. And now the weights of the network are modified as the error is propagated. Following describe the phase for backpropagation learning algorithm.

Phase-1: In Forward propagation of training phase the input pattern's through the neural network in order to generate the propagation's output activations [18]. The first step is initializes the weight vector W_{ij} and W_{jk} and the threshold values for each PE(Processing Element) with minimum random numbers. In the hidden layer, each PE computed the weighted sum according to the equation,

$$net_{aj} = \sum W_{ij} O_{ai} \quad (9)$$

Where, O_{ai} is the input of unit i for pattern number a.

The threshold of each PE was then added to its weighted sum to obtain the activation active(j) of that PE i.e,

$$\text{activ}_j = \text{net}_{aj} + u_{hj} \qquad (10)$$

Where u_{hj} is the hidden threshold weights for j^{th} PE.

This activation determined whether the output of the respective PE was either 1 or 0 by using a sigmoid function,

$$O_{aj} = \frac{1}{1+e^{-k_1 \cdot activ_j}} \qquad (11)$$

Where, K_1 is called the spread factor, these O_{aj} where then served as the input to the output computation.

Signal O_{aj} were then fan out to the output layer according to the relation,

$$net_{ak} = \sum W_{jk}\, O_{aj} \qquad (12)$$

And the output threshold weight uo_k for k^{th} output PE was added to it to find out the activation $activO_k$

$$activO_k = \text{net}_{ak} + \text{u0}_k \qquad (13)$$

The actual output threshold weight O_{ak} was computed using the same sigmoid function which was

$$O_{ak} = \frac{1}{a+e^{-k_1 \cdot activO_k}} \qquad (14)$$

Here, another spread factor K_2 has been employed for the output units.

Phase-2: After completing the feed forward propagation, an error was computed by comparing the output O_{ak} with the respective target t_{ak},

$$\delta_{ak} = t_{ak} + O_{ak} \qquad (15)$$

Now time to update the weight and threshold value according to the error computing value δ_{ak} This error was then used to adjust the weights vector W_{jk} using the equation,

$$\Delta W_{jk} = n_2 K_2 \delta_{ak} O_{aj} O_{ak}(1 - O_{ak}) \qquad (16)$$

Where, $\int^1 activO_k = K_2 O_{ak}(1 - O_{ak})$ the derivation of sigmoid function.

The weight vector W_{jk} was then adjusted to,

$$W_{jk} = W_{jk} + \Delta W_{jk} \qquad (17)$$

For the threshold weight of the output PE, similar equation was employed,

$$\Delta uO_k = n_2 K_2 \delta_{ak} O_{ak}(1 - O_{ak}) \qquad (18)$$

And the new threshold weight equated as,

$$U_{ok} = U_{ok} + \Delta uO_k \qquad (19)$$

Phase-3: Now the adjusted weight vector W_{jk} were feed in the hidden layer to adjust the weight vector W_{ij} and threshold weight uh_j. In this layer, the weight vector W_{ij} was computed by using equation,

$$\Delta W_{ij} = n_1 K_1 O_{ai} O_{aj}(1 - O_{aj}) \sum \delta_{ak} W_{jk} \qquad (20)$$

Where, $\int^1 activh_k = K_1 O_{aj}(1 - O_{aj})$; The weight W_{ij} was then adjusted to,

$$W_{ij} = W_{ij} + \Delta W_{ij} \qquad (21)$$

For the threshold weight of the hidden PE, similar equation was employed,

$$\Delta uh_j = n_1 K_1 O_{aj}(1 - O_{aj}) \sum \delta_{ak} W_{jk} \qquad (22)$$

And the new threshold were calculated,

$$uh_j = uh_j + \Delta uh_j \qquad (23)$$

Repeat previous phase circularly until network is adapted within the minimum error level.

5.6 Recognition

After completing the training process the network weights and thresholds are adaptive according to the number of iterations. So now it is time to perform a recognition process for the new pattern set object. To perform the recognition process for the new pattern object, first it's need take data set from the feature extraction process. Next the data set will be feed for the network as input set. The network takes these value and summation with adapted hidden layer weights and compares this weighted sum with hidden layer threshold.

$$y^H(n) = y_i[w^T(n)x(n)] \qquad (24)$$

Now the hidden layer output is the summation with adapted weights of the output layer.

$$y^U(n) = y_i[w^T(n)x(n)] \qquad (25)$$

Compare those weighted sum into output layer threshold. Here we want to use the activation function for the recognition process is following,

$$y_i(x) = \begin{cases} 0, & z_i < 0 \\ 1, & z_i \geq 0 \end{cases} \qquad (26)$$

After getting the output layer, we calculate the error according to the targeted output in the following error calculating formula,

$$Error_a = 0.5 \sum (t_{ak} - o_{ak})^2 \qquad (27)$$

Where, 'a' is the input pattern, t_{ak} is the targeted output and O_{ak} is the actual output [17].

Here, we can conclude easily the result of the input pattern that is recognized or not.

6 COMPLEXITY ANALYSIS

There are several complexity is associated with the pattern recognition process and also for the Multi-layer Perceptron neural network to implement this process. Here we are presenting some is below,

6.1 Influence the Parameter of MLP

a. **Number of epochs:** Increasing the number of iteration and epochs on the network can be better the performance but sometimes it may be an adverse effect for the wrong recognitions. This partially can be attributed to the high value of learning rate parameter as the network approaches its optimal limits and further weight updates result in bypassing the optimal state. With further iterations the network will try to 'swing' back to the desired state and back again continuously, with a good chance of missing the optimal state at the final epoch. This phenomenon is known as over learning [14].

b. **Number of Input**: It is very simple matter is that if the number of input instance is increased then it required large topological network more training so that it can be influencing on the performance due to increase the error rate. For example, if the input set is 90 symbols then the optimal topology reached 250 hidden layer neurons.

c. **Learning Rate Parameter**: Due to the less value of the learning parameter the network topology links weight are updating slowly and more refined manner. It required more number of iteration in the network to reach its optimal state. So the learning rate parameter variation can be affects the overall network performance.

6.2 Formulation of the problem

The success of any pattern recognition investigation depends to a large extent on how well the investigator understands the problem [15]. Formulation of the problem aims in gaining a clear understanding of the investigation and planning for the remaining stages. Probably it is more difficult to understand and investigation is that the result of the present study and what will be the consequence of the various outcomes. Through the data collection process discrimination, priors and costs problem should be consider to be estimated. Cost will be maximizing if the number of observation is increased and due to that caused it is really difficult to achieve the desired performance from the network.

6.3 Data Collection

Collecting data is important part for pattern recognition design because each stage of the pattern recognition, like feature extraction, training set design and classification are associated with this. A large number of data samples are necessary if the separation between classes is small or high confidence in the error rate estimate is desired [15]. If the separation between data classes is large then its required large number of input data instance to decreased the error rate for a desire network. So to get the better outcome from the network its needed better data collecting method to get desire performance for each stage.

6.4 Primary Inception of Data

The collecting data should be rushed free and analysis using complicated multivariate techniques. To perform inception operation on the data it needs to check the quality of the data, calculating the summary statistics of the data an sketch the plots of the data to get the structure of the data. Primary inception on the data giving a vital clue that analysis should be undertaken; so that it can be saving a lot for the wasted efforts.

6.5 Feature Extraction

Feature extraction measured the data that builds from original data in a way that it should be informative, non-redundant, and subsequent for the learning. So dimensionality reduction the amount the resources to produced set of data that should be feed for the network. This new data may be obtained by a linear or nonlinear transformation of the original set (feature extraction) [15].

7 CONCLUSION

Pattern recognition is the most important field to recognition different real world object pattern by machine with human-like-brain is artificial neural network. As machines capable of automatic pattern recognition have many fascinating uses in science and engineering as well as in our daily lives [18]. According to Watanabe is that "Pattern recognition is a fast-moving and proliferating discipline. It is not easy to form a well-balanced and well-informed summary view of the newest developments in this field. It is still harder to have a vision of its future progress." [20] In order to have the best opportunity of developing effective solutions, it is important to adopt a principled approach based on sound theoretical concepts [19]. So in this paper we are trying to present the theoretical representation of pattern recognition design process using Multi-Layer Perceptron that is the supervised learning method in the branch of artificial neural network which can be express the result is like a human brain.

REFERENCES

[1] E. Backer, "Computer-Assisted Reasoning in Cluster Analysis" Prentice Hall, 1995.

[2] B. Ripley, "Statistical Aspects of Neural Networks", Networks on Chaos: Statistical and Probabilistic Aspects. U. Bornndorff-Nielsen, J. Jensen, and W. Kendal, eds., Chapman and Hall, 1993.

[3] Violeta sandu, Florin Leon, "RECOGNITION OF HANDWRITTEN DIGITS USING MULTILAYER PERCEPTRONS", Universitatea Tehnică „Gheorghe Asachi" din Iaşi Tomul LV (LIX), Fasc. 4, 2009.

[4] Anil K. Jain, Fellow, IEEE, Robert P.W. Duin, and Jianchang Mao, Senior Member, IEEE, "Statistical Pattern Recognition: A Review" IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, january 2000, pp. 4-37.

[5] S. Watanabe, "Pattern Recognition: Human and Mechanical" New York: Wiley, 1985.

[6] Bishop, Christopher M. (2006), "Pattern Recognition and Machine Learning" Springer. p. vii.

[7] http://aitopics.org/topic/pattern-recognition, - Pattern Recognition Laboratory at Delft University of Technology

[8] R. Picard, Affective Computing. MIT Press, 1997.

[9] En Wikipedia, “Pattern Recognition” June 17, 2015.

[10] En Wikipedia, “Multilayer Perceptron”, June 22, 2015.

[11] En Wikipedia, “Backpropagation”, June 22, 2015.

[12] Jayanta Kumar Basu, Debnath Bhattacharyya, Tai-hoon Kim, "Use of Artificial Neural Network in Pattern Recognition" , International Journal of Software Engineering and Its Applications, Vol. 4, No. 2, April 2010.

[13] Tin Kam Ho, Mitra Basu and Martin Hiu Chung Law , "Data Complexity in Pattern Recognition", ISBN 978-1-84628-172-3 , 2006

[14] Daniel Admassu, "Unicode Optical Character Recognition", Codeproject.com in 23 Aug 2006.

[15] Andrew R. Webb, "Statistical Pattern Recognition", ISBNs: 0-470-84513-9 (HB); QinetiQ Ltd., Malvern, UK.

[16] R Beale and T jackson, “Neural Computing: An introduction”, Adam Hilger, Bristol, Philadelphia and New York.

[17] Md. Rabiul Islam and Kaushik Roy, “An Approach to Implement the real time eye recognition system using artificial neural network”, Proceeding of the ICEECE, December 22-24, Dhaka, Bangladesh.

[18] Tin Kam Ho, Mitra Basu and Martin Hiu Chung Law, "Data Complexity in Pattern Recognition", ISBN 978-1-84628-172-3, 2006.

[19] Christopher M. Bishop, "Neural Networks for Pattern Recognition", Clarendon Press-OXFORD, 1995.

[20] S. Watanabe, ed., “Frontiers of Pattern Recognition” New York: Academic Press, 1972.

[21] "View-invariant action recognition based on Artificial Neural Networks", IEEE 2012 Transactions on Neural Networks and Learning Systems, Volume: 23 , Issue: 3.