A Study on Artificial Intelligence in the Diagnosis of Fault Mechanism System

Mr. Lalit Singh Bisen[1] Dr. Dharmendra kumar singh[2]
[1]M.Tech. Scholar [2]Associate Professor
[1,2]Dr. C. V. Raman Institute of Science & Technology

***Abstract*— *An important application of artificial intelligence (AI) is the diagnosis of faults of mechanisms and systems in general. A traditional approach to the problem of diagnosis is to construct a heuristic, rule-based system which embodies a portion of the compiled experience of a human expert. These systems perform diagnosis by mapping fault symptoms to generated hypothesis to arrive at diagnostic conclusions.*

Keywords— Localization, Wireless Sensor Network, Artificial Intelligence

I. INTRODUCTION

In power system engineering the pattern recognition plays very important role. The knowledge acquisition and search process involved in expert systems is exhaustive and hence time consuming. In addition, the simulation of models is usually too slow to be effectively applied in a real-time environment. Artificial Neural Networks (ANN) is found to be suitable for the above requirements.

They are massively parallel interconnected networks of simple adaptive elements and their Hierarchical organizations, which are intended to interact with the objects of the real world in the same way as the biological counterparts. Neural networks find wide applications in Parallel distributed processing and in real-time environments. Neural Networks have considerable advantages over expert systems in terms of knowledge acquisition, addition of new knowledge, performance and speed.

Recently, interest in the application of associative memories and neural networks to problems encountered in diagnostic expert systems development has increased. Neural Networks appear to offer features which coincide well with the requirements of pattern-based diagnosis. An important feature of fault diagnosis using neural networks is that they can interpolate the training data to give an appropriate response for cases described by neighboring or noisy input data. This paper describes the design and simulation of a neural network for fault detection and diagnosis of power systems. In this paper fault diagnosis is conceptualized as a pattern classification problem which involves the association of patterns of input data representing the behavior of the power system to one or more fault conditions. The neural network is trained off-line with different fault situations and used on-line. The diagnostic system was able to detect and diagnose the faulted component corresponding to the input pattern consisting of switching status of relays and circuit breakers.

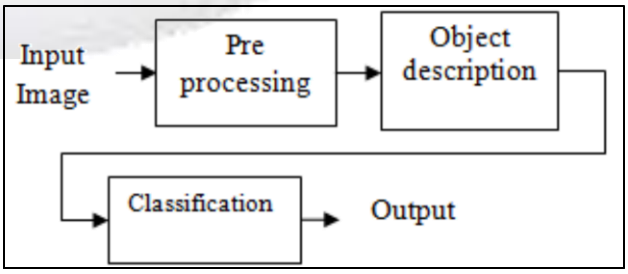
A. Neuron Computing: A Pattern Recognition Perspective

Pattern recognition could be formally defined as categorization of input data into identifiable classes via extraction of significant features or attributes of the data from the background of irrelevant detail. A pattern class is a category, determined by some given common attributes. When a set of patterns falling into disjoint classes are available, it is desired to categorize these patterns into their respective classes through the use of some automatic device. It is important to note that learning or training takes place only during the design (or updating) phase of a pattern recognition system. Once acceptable results have been obtained, the system is engaged in the task of actually performing recognition on samples drawn from the environment in which it is expected to operate. The main objective of this research is to investigate and develop a general approach formally from a new theory based on Structural or Syntactic Pattern Recognition.

The problem of pattern recognition is divided into the following sub problems:

1) Image pre processing
2) Object Description
3) Classification.

The pre-processing steps involve noise removal, conversion of gray scale and colour images into binary images and the extraction of contour from the binary image. The object description module takes an input contour image and gives a vector of direction and length called a knowledge vector as output. This plays an important role in the whole process as the knowledge vector gives more information about the objects present in the image which is used to characterize the pattern. We use a syntactic approach for the description of objects. The knowledge vector obtained here gives information about the direction and the length of lines in the contour of objects which can then be given as input to a classifier module. The normalization of this vector plays an important role in the classification process. The vector has to be analyzed and normalized in such a way that it better suits for more variance of objects. The classification module takes the normalized vector as input and identifies them as a member of one of the predefined classes depending on the set of attributes that they hold. The design of a classifier involves the selection of the classifier and the estimate of parameters for the classifier. The feed forward artificial neural networks with back-propagation learning algorithm could be used for classification. Details of such neural networks could be obtained from standard literature.



II. Pre-processing

To improve the quality of the noisy images we get from the real time environment image enhancement and restoration techniques can be used. Pixel neighbor processing techniques such as low pass filtering, high pass filtering or median filtering are used for the removal of noise in the image. The pixel point processing techniques such as binary contrast enhancement, stretching or shrinking and histogram equalization are used to improve the contrast of the image.

The input image to the system may be a colour image, a gray scale image or a binary image. These images have to be converted to binary images for further processing. Segmentation is an important process in our object recognition system as it highlights the objects present in the image from its background. The function im2bw() in matlab is used to convert the images to binary. The graythresh() function computes a global threshold (level) that can be used to convert an intensity image to a binary image with im2bw. Level is a normalized intensity value that lies in the range [0, 1]. The contours are extracted as they give the outlines of shapes of the objects present in the image. The function bwtraceboundary() in matlab is used for finding the contours. The initial point of the object is identified, and the outline tracing algorithm is used. bwtraceboundary() traces the outline pixels of an object in binary image. Nonzero pixels belong to an object and zero pixels constitute the background. Figure 2 shows a sample binary image and its contour.

A. Object description

The notion of 'Picture Description Languages' refers to the language of digital images defined over a regular array of vertices. Having extracted the contour of an image this process involves running through the contour pixel by pixel and performing some form of calculations using every pixel and its surrounding pixels. This involves finding the length of the contour in every possible direction. The eight possible directions are two horizontal directions, two vertical directions and four diagonal directions. As the lines other than straight lines have more variations in their directions, the length of the descriptor becomes large. It is then reduced to a single occurrence of eight directions and normalized to fixed number of pixels for making it invariant. Following is the knowledge vector of a hexagon. So, the final vector is a normalized vector consisting of a single occurrence of eight directions with their length, which can then be fed into an artificial neural network for further process of classification.

B. Artificial Neural Networks

Learning from a set of examples is an important attribute needed for most pattern recognition systems. Artificial neural network is an adaptive system being widely used in pattern recognition systems that changes its structure based on external or internal information that flows through the network. Artificial neural networks are adjusted, or trained, so that a particular input leads to a specific target output. The neural network design part consists of two processes, training and application. The training of the neural network continues until the mean squared error reduces to a certain threshold or until the maximum number of iterations is reached. Once training is completed, the network can be applied for the actual classification of the data.

The classification technique used may be one of the following:

1) Supervised classification - in which the input pattern fall as a member of a predefined class.
2) Unsupervised classification - in which the pattern falls into an unknown class as there are no predefined classes.

The learning or training takes place only during the design phase of a pattern recognition system. Once the results obtained are satisfactory, the system is ready to perform the task of recognition on samples drawn from the environment in which it is expected to operate.

C. Feed-forward neural networks

Feed-forward networks are commonly used for pattern recognition. A three-layer feed forward neural network is typically composed of one input layer, one output layer and one hidden layer. In the input layer, each neuron corresponds to a given input pattern while in the output layer each neuron corresponds to a predefined pattern. Once a certain sample is input into the network, he best situation is that the output will be a vector with all elements as zero only except the one corresponding to the pattern that the sample belongs to. Of course, it is very complex to construct such types of neural networks. The commonly used networks for minimizing the cost are multi-layer-feed-forward neural networks, which uses the back-propagation learning algorithm for training neural networks.

D. Back-Propagation Algorithm

Multi-layer feed-forward networks have been used as powerful classifiers. The training of backpropagation algorithm involves 4 stages

1) Initialization of weights
2) Feed forward stage
3) Back propagation of errors
4) Updating of weights and bases

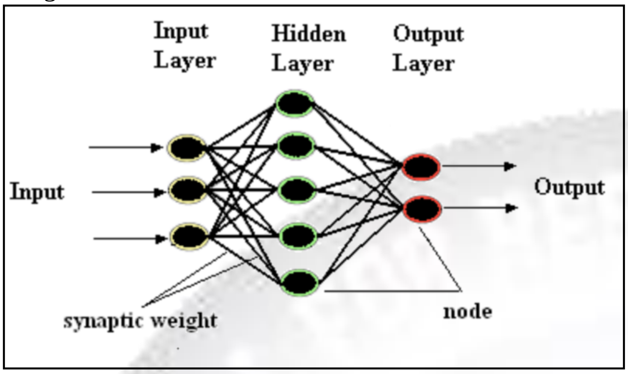
During the feed-forward stage each layer in the network calculates its activation value and passes it to the layers in the next level. The neurons in the output layer produce the output of the network and compare it with the target output to determine the error. During the back-propagation stage the error is back propagated from the output layer to each layer in the previous level for the correction of the adjustable parameters. This process repeats until the error reaches a minimum threshold or until the maximum number of iterations performed. The Back propagation algorithm is discussed below.

1) Parameters used

- $x = (x_1, x_2 \ldots\ldots, x_i, \ldots., x_n)$ - Input training vector
- $t = (t_1, t_2, \ldots\ldots.., t_k, \ldots., t_m)$ -Output target vector
- δ_k= error at output unit y_k
- δ_j = error at hidden unit z_j
- α= learning rate
- v_{ij}= weights of input layer
- v_{oj}= bios on hidden unit j
- z_j= activation of hidden unit j
- w_{jk} = weights of hidden layer
- w_{ok}= bios on output unit k
- y_k= activation of output unit k

III. Multilayer perceptron

In ANNs, neurons are arranged into groups called layers. The term Multilayer Perceptron [1] is self- explanatory, multi means more than one. Thus, Multilayer Perceptron has morethan one layer of neurons in its network as shown in the figure



The description of each layer is given as under:

A. Input layer:

It consists of a set of neurons equal to no. of input variables that receive inputs form the external environment. In this layer, there is no activation function; no processing of input variables and output of each neuron is the same as its input.

B. Hidden Layer:

In any Multilayer Perceptron neural network, there can be a number of hidden layers between the input and output layers. It is mathematically proven fact that a single hidden layer is sufficient to approximate any function to any degree of accuracy provided there is enough number of neurons in the hidden layers. The number of neurons in a hidden layer is chosen very carefully to limit the complexity and at the same time to achieve better functional approximation accuracy. The function approximation accuracy improves with increase in the neurons in the hidden layer, but, the network complexity increases and the response time of network deteriorates. Both the factors are important and act contrary to one another. In order to maintain balance between the two opposing constraints, the judicious choice of the number of hidden layer neurons becomes very significant.

C. Output Layer:

The output layer consists of neurons that communicate the output of the system to the user or the external environment. The number of neurons in the output layer is equal to the number of output variables. The neurons in this layer receive their input from the preceding layer or last hidden layer in the network.

IV. Conclusion

In this Paper a syntactic approach is proposed for shape based object recognition. The knowledge vector is reduced to the input vector to a neural network by some vector approximation and normalization processes. The research efforts during the last decade have made significant progresses in both theoretical development and practical applications. The method presented here may offer a promising solution for object recognition problem. The model power system employed for fault diagnosis. Artificial neural networks are associated with the inherent properties of Learning, Generalization, Abstraction and Applicability and hence its performance is measured in terms of the above features. ANNs modify their response to the environment. Given a set of inputs along with desired outputs, they self organize to produce consistent properties. Once trained, the networks response can be insensitive to minor variations in its input. ANN generalizes automatically as a result of its structure. ANNs are capable of abstracting the essence of a set of inputs. They can be trained to produce something it has-not seen before. They are best suited for large class of pattern recognition tasks through the Connectionist property.

References

[1] Nilsson, N. J ‘LEARNING GA CHINE’,S Mc Graw-Hill, NY,1965.
[2] Fu,I<. S ‘SEQUENTIAL TIME IN PATTERN RECOGNITION AND MACHINE LEARNING A’c, ademic Press, NY 1968
[3] Rosenblatt, ‘PRINCIPE OF EURODYNAMICSS patan,1962.
[4] Minsky.M and Pappert, ’PERCEPTROMNIST’, Press, Canibridge,1969.
[5] Duda, R. 0 and P.E.Hart, ‘PATTERN C ASSIFICATION AND DESIGN ANALYSIS’, iley , NY, 1973.
[6] Michalaski. R. S , J.S Carbonell and T.M.Mitche1, ‘MACHINE LEARNIN:GA N ARTIFICIAL INTELLIGEN APPROACH T Publications 1983.
[7] Denker.S ‘NEURAL NETWORKSF OR COMPUTING A’IP. Conference Proceedings, No. 151. NY, 1986.
[8] Anderson, J.A and Rosenfield E L N ~ ~ ~ t ~ - C ~ ~ FOUNDATION OF RESEARCH’, IT Press, Canibrirlge, Mass,1988.
[9] Wasserman, P. D ‘NEURAL COMPUTING THEORY AND PRACTICE’V, an Nostrand, Reimhold, NY, 1989.
[10] Pao, Y. H ‘ADAPTIVE PATTERN RECOGINITION AND NEURAL NETWORK,S A’ ddison Wessly, NY, 1989.s