*Mathematical & Computational Applications*, Vol. 8, No. 2, pp. 181-189, 2003


# A NEURAL NETWORK IMAGE RECOGNITION FOR CONTROL OF MANUFACTURING PLANT

Bekir Karlik
University of Bahrain, College of Information Technology,
Department of Computer Engineering, P O Box 32038, Bahrain
bkarlik@itc.uob.bh

**Abstract-** The aim of this work is to interpret the image for vision-based logic control of manufacturing plant. The images, which had been taken by an overhead fixed camera, were transformed with feature value by image preprocessing. A multi-layered neural network was used to recognize randomly selected images of metallic pegs and plastic rings. Images, which are not recognized by the neural network, will be rejected and consequently the actuator will allow the objects to pass through. On the other hand, if the images are recognized by the neural network then the logic controller generates different type of action and the actuator pushes the object down.
**Keywords** – Image processing, neural network

## 1. INTRODUCTION

Neural network modeling approaches are currently attracting significant interest in any industrial applications (one example being systems control and analysis). In brief, they provide powerful empirical modeling techniques for complex data where the inter-relationships are not fully understood or amenable to exact analytical solutions. Such data may be non-linear, inter-dependent, noisy and non-systematic. It has been demonstrated that neural network techniques provide significant improvements over linear regression analysis in various real physical systems and there is increasing interest in their applicability to materials modeling. Neural networks have been used to control process and robotic, predicting the properties of the product from the raw materials, finding the dynamics of a projectile moving with air friction, solving problems of vibration of a beam-mass system [1-5]. They have been applied to similar manufacturing applications by other workers, e.g. to control work-in-process inventory in a manufacturing line and to process parts loading scheduling in a flexible forging machine. [6-7]. Neural networks have also been used to recognize and understand images for different purposes. Typically the output data from the testing and instrumentation used is often difficult to interpret and large in quantity, making it an ideal candidate for neural network interpreting [8–11]. These studies have resulted in various expert systems for image processing and image understanding. Image analysis is a process of discovering, identifying, and understanding patterns that are relevant to the performance of an image-based task.

Traditionally, computer vision systems have separated the object recognition task into two independent subtasks: a- feature extraction and b- classification. The feature extraction task begins with an object and a procedure for extracting relevant features. The features are chosen by the system designer to be invariant to the object's position, scale, and orientation. The output of this task (which is a vector of feature values) is then passed to the classification subtask. Based on large set of these vectors, the classifier determines which are the distinguishing features of each objects class such that new vectors are placed into the correct class within a predetermined error [12].

A more recent approach for distorting invariant object recognition combines the tasks into a single system. Given only a set of views of each object class, the system determines which features to extract as well as which are the distinguishing features of each class. The advantage of this approach is that the two subtasks can share information and improve the classifiers` separating ability by extracting the useful features. The disadvantage, however, is that the system requires a longer training period since it has no prior information about the relationship between the set of training views. Object recognition systems based on neural networks are an example of above mentioned. Recently, it has been shown that neural networks have abilities to solve various image recognition problems, especially, the multi-layered feed-forward networks, which have a better ability to learn the correspondence between input patterns and teaching values from many sample data by the error back-propagation algorithm [13]. Thus the three-layered feed-forward neural network which, was used in this work, has been trained by error back-propagation.

## 2. BACKGROUND

### A-Image Processing and Analysis

The Computer Vision System has been used in many industrial applications very successfully to solve various problems for inspection and control areas. The computer vision can be defined to interpret and characterize 2 or 3 dimensional images. This is also known robot vision, artificial vision and machine vision. The common characteristics of these systems are as follows:

- Natural or artificial (like the laser) lighting
- Multiple sensing systems, for example camera
- Image processing
- Filtering and Windowing
- Feature extraction by using different algorithms
- Pattern recognition, comprising of the past images

Scene and object illumination plays a key role in the computer vision process. The central purpose of imposing controlled constant illumination is to enhance visually the parts to be imaged so that their flaws, defects, and features are highlighted and so that their identification and classification by the vision system becomes somewhat easier [9]. The image has to be converted or digitized to process further. This process is called "the Image digitization". In this process, every pixel of image has to be divided into Small Square and each square is saved into the memory as an integer number. In that way, the brightness of picture is 8 bit, which contains 256 gray levels. Good digitization will produce good resolution, but high-resolution picture requires high memory computer facilities [14].

Image processing can be thought of as a transformation, which takes an image into an image, i.e., it starts with an image and produces a modified (enhanced) image. To perform this task within very short period of time causes some problem. These problems can be avoided as follows:

- Data reduction
- Segmentation
- Feature extraction
- Object recognition

**Data reduction:** The purpose of this process to reduce space in the memory, to do that digital conversion and windowing are used.

**Segmentation:** It is used to describe a grouping process in which the components of a group are similar with respect to some feature or set of features. Determination of the divided segments with similarity (boundaries and regions) is used to reduce data. There are various segmentation methods. Some of the important methods are; threshold, region growing and edge detection.

**Feature extraction and object recognition:** It is used mostly object recognition and also it helps to reduce data. There are various methods for this subject such as Template matching, Hough transform, neural network etc.

### B. Artificial Neural Networks (ANN)

Artificial neural networks (ANNs) are computational tools loosely based on an understanding of the structure and function of the brain. An ANN is composed of many simple inter-connected nodes (neurons) where the interconnections are 'weighted' and these weights are adapted to improve it is overall performance. Such networks are generally applied to modeling or classification tasks where a considerable amount of data are available which adequately describe the characteristics of the underlying relationship, and the success of such an application is often dependent on the quality of the data as well as the expertise of the designer in selecting an appropriate network structure. Rumelhart, Hinton, and Williams [15] dealing with the development of new training algorithms for multilayer perceptron have changed matters considerably. Their basic method, often called the "generalized delta rule for learning by back-propagation", provides an effective training method for multilayer machines. Although this training algorithm cannot be shown to converge to solution in the sense of the analogous proof for the single-layer perceptron, the generalized delta rule has been used successfully in various problems of practical interest. This success has established multi-layer perceptron as one of principal models of neural networks currently in use. The back-propagation-trained multi-layer neural networks are the most popular method used neural network-based object recognition systems. This method is illustrated in Figure 1. It is composed of layers of simple computational elements (called neurons or nodes), which imitate the most basic function of a biological neuron. Each layer receives inputs from the previous layer and thus information flows through the network in a feed-forward fashion. The training process consists of applying input vectors sequentially and adjusting the network weights using a gradient descent-learning rule until the input vectors produce the desired output vectors within some predetermined error.

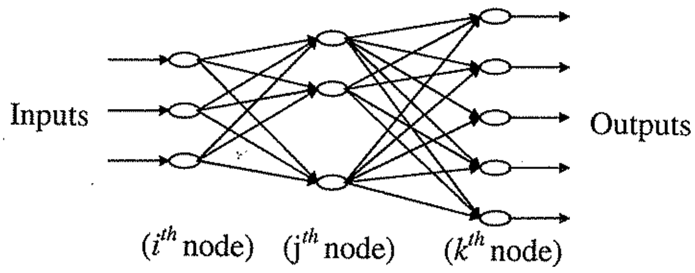


Figure 1. Example of an MLP

To compute a node's output ($O_j$ for unit j) a weighted summation of the incoming input signals is taken:

$$O_j = f(net_j)\ ,\ \ net_j = \sum_i w_{ji} O_i + \theta_j \qquad (1)$$

Where $O_i$ is the output of unit i, $w_{ji}$ is the weight of the connection from unit i to unit j, $\theta_j$ is the bias of unit j, $\Sigma_i$ is a summation over every unit i whose output flows into unit j, and $f(x)$ is a monotonously increasing function. In practice, a logistic activation function (sigmoid function) $f(x) = 1/(1+\exp(-x))$ is used. When the set of m-dimensional input patterns $\{i_p = (i_{p1}, i_{p2}, \ldots, i_{pm})\ ;\ p \in P\}$ where P denotes set of presented patterns, and their corresponding desired n-dimensional output patterns $\{t_p = (t_{p1}, t_{p2}, \ldots, t_{pm})\ ;\ p \in P\}$ are provided, the neural network is trained to output ideal patterns as follows. The squared error function $E_p$ for a pattern p is defined by

$$E_p = \frac{1}{2}\left[\sum_{j \in output} (t_{pj} - o_{pj})^2\right] \qquad (2)$$

$t_{pj}$ : target (desired) value, $o_{pj}$ : actual network output value.

The purpose is to make $E = \Sigma_p E_p$ small enough by choosing appropriate $w_{ji}$ and $\theta_j$. To realize this purpose, a pattern $p \in P$ is chosen successively and randomly, and then $w_{ji}$ and $\theta_j$ are changed by

$$\Delta_p w_{ji} = -\varepsilon\,(\partial E_p / \partial w_{ji}) \qquad (3)$$
$$\Delta_p \theta_j = -\varepsilon\,(\partial E_p / \partial \theta_j) \qquad (4)$$

Where $\varepsilon$ is a small positive constant. By calculating the right hand side of (3) and (4), it follows that

$$\Delta_p w_{ji} = \varepsilon\, \delta_{pj} O_{pi} \qquad (5)$$
$$\Delta_p \theta_j = \varepsilon\, \delta_{pj} \qquad (6)$$

Where

$$\delta_{pj} = \begin{cases} f'(net_j)(t_{pj} - O_{pj}) \\ f'(net_j)\sum_k w_{kj}\delta_{pk} \end{cases} \tag{7}$$

Note that k in the above summation represents every unit k in the layer following the layer of j (unit j). In order to accelerate the computation, the momentum terms are added in (5-6),

$$\Delta_p w_{ji}(n+1) = \varepsilon\, \delta_{pj} O_{pi} + \alpha\, \Delta_p w_{ji}(n) \tag{8}$$

$$\Delta_p \theta_j(n+1) = \varepsilon\, \delta_{pj} + \alpha\, \Delta_p \theta_j(n) \tag{9}$$

Where n represents the number of learning cycles, and $\alpha$ is a small positive value. In this study, by trial and error the optimum $\alpha$ and $\varepsilon$ constant values were determined to be: $\alpha = 0.7$, $\varepsilon = 0.9$.

## 3. APPLICATION AND RESULTS

In this work, it was proposed a novel technique for vision-based logic control of manufacturing plant. The logic control of manufacturing system, shown in Figure 2 (under the block diagram), represents a component sorting and assembly process that can be controlled by virtually any PLC. The upper conveyor motor and the lower conveyor motor drive the upper conveyor and the lower conveyor respectively. A random selection of metallic pegs and plastic rings are placed on the upper conveyor. The rings and pegs need to be identified and separated. A camera does this. By means of the sort solenoid (up actuator), plastic rings can be ejected down the assembly chute, which can have up to five plastic rings. Metallic pegs, meanwhile, continue on the upper conveyor and are deflected down the feeder chute. The feeder chute automatically feeds pegs onto lower conveyor. The camera also is used to determine whether or not the assembly area is empty. If it is, the assembly solenoid (down actuator) is used to dispense a ring from the assembly chute into the assembly area. The assembly area is positioned just above the lower conveyor and, when a metallic peg passes, the peg engages with the hole in the ring and the two components are assembled. The lower conveyor is used to control the process, and a PC-based package called 'Quadriga' is used to program the PLC [16].

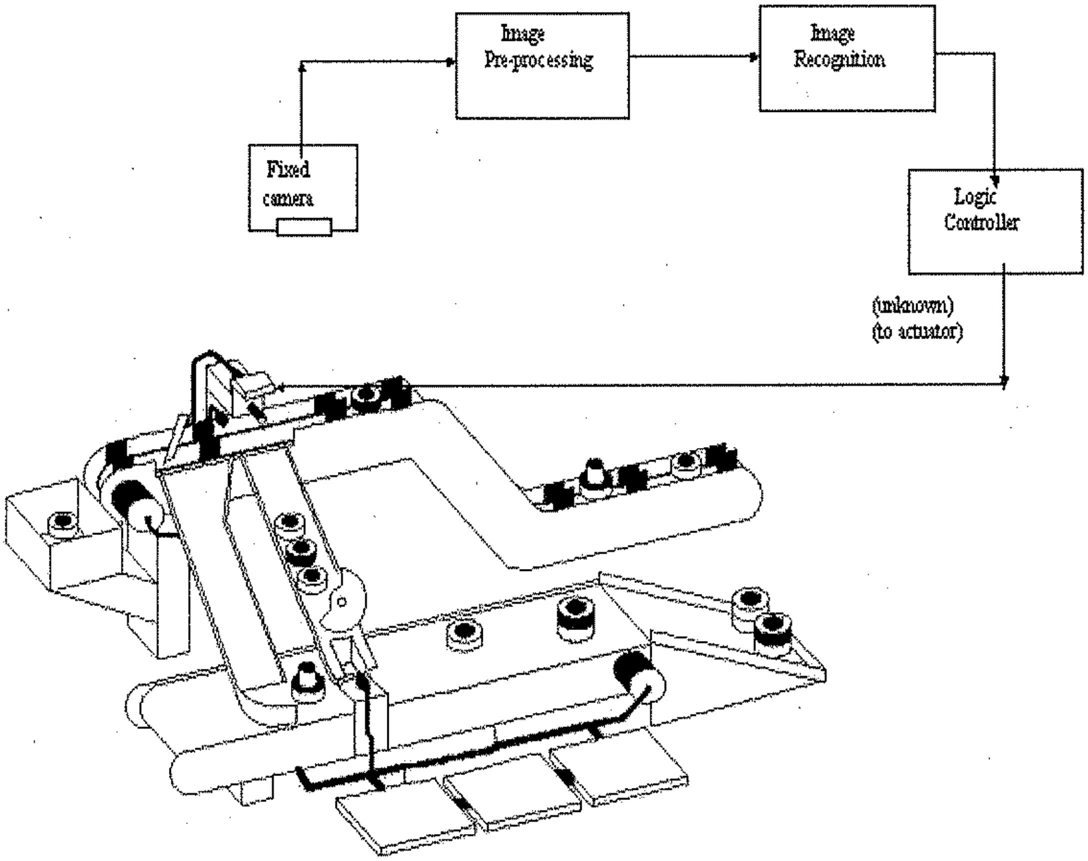


Figure 2. Image Recognition for vision-based logic control of manufacturing plant

However, Figure 2 describes the block diagram of an image recognition processing. Firstly, an overhead fixed camera are taken the images that are collected in a file had taken the pictures. Then each image is converted 256 gray levels as you can see in Figure 3. In order to recognize the image adequately, the whole image is split into 100x100 pixels and divided bmp data, which is used as a raw data. 1078 characters are redundant at the beginning of each bmp file. The software reads these redundant data, and then the rest of the data is recorded.

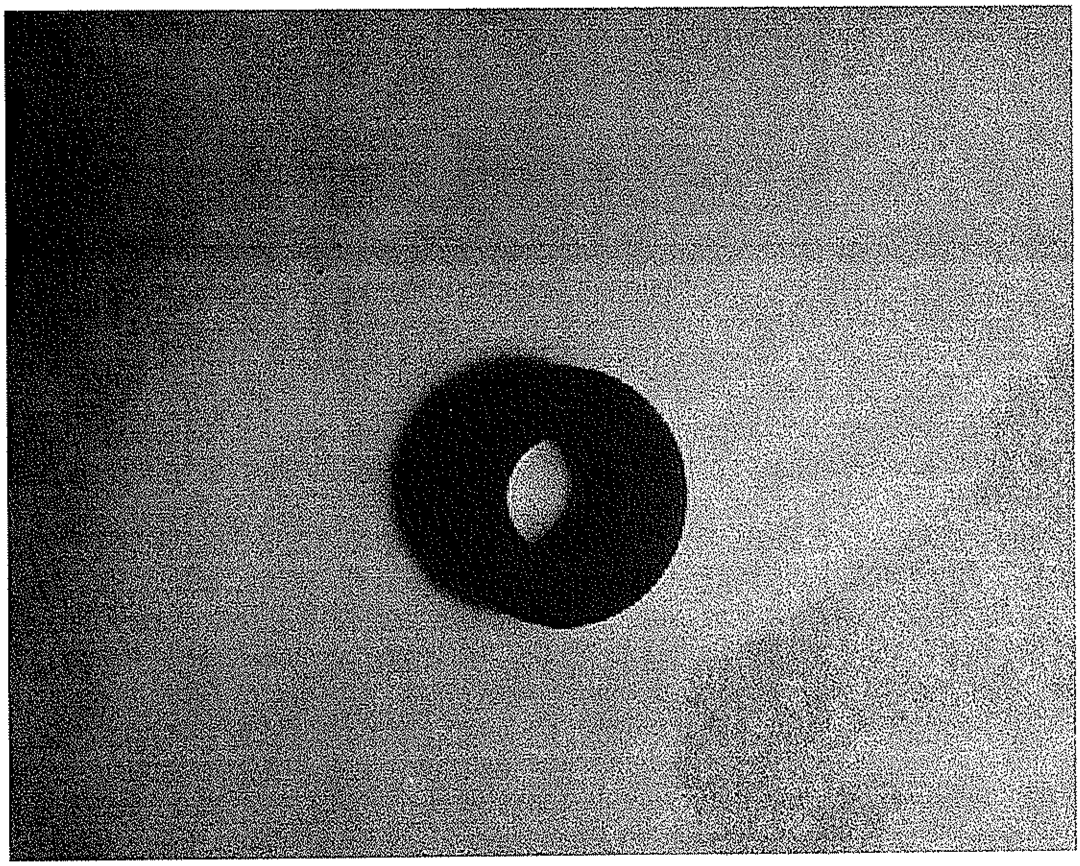

Figure 3. A sample camera image

In this work, reduced resolution was achieved by means of a simple square block. 100x100-pixel image was divided 5x5 square block. The average of these 5x5 pixels was taken as a 1 pixel. It was the training set which was constructed by taking a 5x5 pixel for each 100x100 pixel images. Therefore, 10Kbyte long unnecessary information was ignored then 100x100-pixel image reduced to 20x20 pixels. Having saved with dot extension, then image was sent to screen. Finally, the saved data was used as an input to neural network. The output of neural network is used to send information to logic controller. The logic controller generates different type of action about the actuator using this information.

The training time took about 16 minutes for 1000 iteration. Figure 4 shows some samples of testing images. The architecture of neural network contains three layers, first layer 400 nodes, second layer 10 and output layer 4 nodes. The output of neural network was logical output as 1000, 0100, 0010, and 0001 for four nodes. When all of the test data were introduced neural network, they were recognized successfully. We found optimum learning rate (0.9) and momentum coefficient (0.7) after the different iterations.

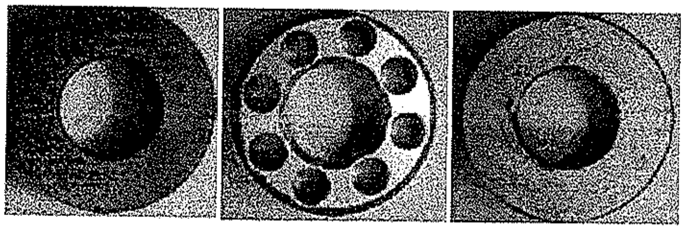

Figure 4. Samples images for testing

## 4. CONCLUDING REMARKS

It has been shown that neural computing methods may be successfully applied to the manufacturing plant control. A design of a discrete event control systems for a manufacturing system is considered: A Control Petri-net is developed for this manufacturing system and then the Token Passing Ladder Logic methodology is used to derive structured ladder logic code depending on the vision based interpreter by artificial neural network. The output of neural network was logical output obtained from nodes of the last layer. When all of the test data were introduced neural network, they were recognized successfully. It can be seen Figure 4, some of these images are the same size and shapes. But their gray levels are different. So if a sensor uses instead of a camera, it is impossible to recognize the images. But it is possible to use different structures of neural networks such as convolutional neural networks to control of the manufacturing plant. In addition the advantages of the structure are raw image usability, noise resistance, shift invariant and distributed feature implementation.

## REFERENCES

1. N. Kubota and T. Fukuda, Structured Intelligence for Self-Organizing Manufacturing Systems, *Journal of Intelligent Manufacturing*, **10**, no. 2, pp. 121-133, 1999
2. B. Karlik and S. Aydin, An Improved Approach to the Solution of Inverse Kinematics Problem for Robot Manipulator, *Engineering Applications of Artificial Intelligence*, **13**, pp. 159-164, 2000
3. C. Meric, N.S. Koksal, B. Karlik, An Investigation of Deep Drawing of Low Carbon Steel Sheets and Applications in Artificial Neural Networks, *Mathematical & Computational Applications*, **2**, no. 3, pp. 119-125, 1997
4. M. Pakdemirli, E. Ozkaya, B. Karlik, A Neural Network Algorithm For Dynamics of A Projectile Moving With Air Friction, *Proceedings of the International Conference On Modeling & Simulation* (MS'2001), pp. 220-223, May 23-26, 2001, Lviv, Ukraine
5. B. Karlik, E. Ozkaya, S. Aydin, M. Pakdemirli, Vibration of a Beam-Mass System Using Artificial Neural Networks, *Computer & Structures*, **69**, pp. 339-347, 1998
6. M. Col and B. Karlik, An Artificial Neural Network Case Study: The Control of Work-in-Process Inventory in a Manufacturing Line, *Proceeding of the IEEE International Symposium on Industrial Electronics*, (ISIE'97), **1**, pp.7-11, July 7-11, 1997, Guimaraes, Portugal

7. Y. Tsujimura and M. Gen, Parts Loading Scheduling in a Flexible Forging Machine Using an Advanced Genetic Algorithm, *Journal of Intelligent Manufacturing*, **10**, no. 2, pp. 149-159, 1999
8. Zrimec and C. Sammut, A Medical Image-Understanding System, *Engineering Applications of Artificial Intelligence*, **10**, no.1, pp. 31-39, 1997
9. Adeli and S. L. Hung, Fuzzy Neural Network Learning Model for Image Recognition, *Integrated Computer-Aided Engineering*, **1**, no. 1, pp. 43-55, 1993
10. B. Karlik, S. Aydin, Pattern Recognition by Using ANN for Tactile Sensor of a Robot Manipulator, *Proceedings of the First Symp. on Mathematical & Computational Applications*, pp. 86-90, Nov. 19-21, 1996, Manisa, Turkey
11. S. W. Lee and H. H. Song, A New Recurrent Neural Network Architecture for Visual Pattern Recognition, *IEEE Trans. on Neural Networks*, **8**, no. 2, pp. 331-340, 1997
12. Yoh-Han Pao, *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley Publishing Company, Inc. 1992
13. S. Pandya, R. B. Macy, *Pattern Recognition with Neural Networks in C++*, CRC Press, IEEE Press, Inc. 1996
14. R. C. Gonzales and R. E. Woods, *Digital Image Processing*, Addison-Wesley Publishing Company, Inc. 1993
15. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Parallel Distributed Processing Explorations in the Microstructures of Cognition*, **1**, (Eds.) D.E. Rumelhart and J. L. Mc Clalland, MA: MIT Press, Cambridge 1986, p. 318-362
16. B. Karlik, An Image Interpreter for Vision-Based Logic Control of Manufacturing Plant, *Technical Report, AMME-98-7*, 1998, Salford, UK
17. User Manual *VideoPort MRT*, version 1.0, April, 1994