International Journal of Reliability and Applications
Vol. 10, No. 1, pp. 1-15, 2009

Improvement of an Early Failure Rate By Using Neural Control Chart

K. Y. Jang
Tech Center General Motors Corporations Warren, MI 48092, USA

C. J. Sung
Department of GIS Engineering, Namseoul University, Korea

I. S. Lim*
Department of Industrial & Management Engineering, Namseoul University, Korea

Abstract: Even though the impact of manufacturing quality to reliability is not considered much as well as that of design area, a major cause of an early failure of the product is known as manufacturing problem. This research applies two different types of neural network algorithms, the Back propagation (BP) algorithm and Learning Vector Quantization (LVQ) algorithm, to identify and classify the nonrandom variation pattern on the control chart based on knowledge-based diagnosis of dimensional variation. The performance and efficiency of both algorithms are evaluated to choose the better pattern recognition system for auto body assembly process. To analyze hundred percent of the data obtained by Optical Coordinate Measurement Machine (OCMM), this research considers an application in which individual observations rather than subsample means are used. A case study for analysis of OCMM data in underbody assembly process is presented to demonstrate the proposed knowledge-based pattern recognition system.

Keywords: Back propagation, Learning vector quantization, Knowledge-based

1. INTRODUCTION

Artificial neural network is a computational structure inspired by the study of biological neural processing. Artificial neural network models had been rapidly developed in the area of speech and image recognition. Several algorithms are applied as a new data analysis tool due to those adaptive nature and fast computational capability. Unlike statistical approach, neural network methods are non-parametric in the sense that

---

* Corresponding Author.
E-mail : ikslim@nsu.ac.kr

functional form need not be specified a priori. Rather than relying on a pre-specified functional form, neural networks build their own model by learning, testing and modifying.

Artificial neural networks are computing system coordinating a number of interconnected processing elements called neurons. Initially, arbitrary values can be assigned to the weights of the network. Each case from a sample can be loaded on to the input layer of the network and the input nodes simply send these values to output nodes. Each output node calculates the weighted sum of the inputs using the weighted assigned to the connections. The output or activation value of a neuron is determined by transfer function and the weight values are adjusted by a specified learning rule. The system is inherently parallel in the sense that many units can carry out their computations at the same time.

As shown Figure 1.1, artificial neural networks are computing systems containing a number of interconnected processing elements called neurons. Initially, arbitrary values can be assigned to the weights of the network. Each case from a sample can be loaded on to the input layer of the network and the input nodes simply send these values to output nodes. Each output node calculates the weighted sum of the inputs using the weights assigned to the connections. The output or activation value of a neuron is determined by transfer function and the weight values are adjusted by a specified learning rule. The system is inherently parallel in the sense that many units can carry out their computations at the same time.

One of the most significant attributes of a neural network is its ability to learn by interacting with its environment or with an information source. Learning in a neural network is normally performed through an adaptive procedure, namely learning rule or algorithm. The purpose of the learning rule is to train the network to perform some tasks.

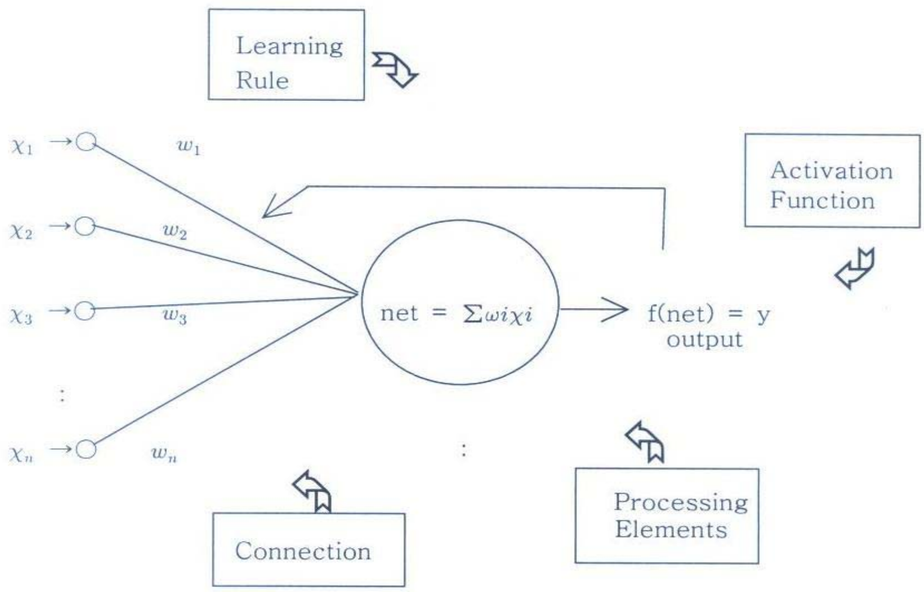

Figure 1.1. Four basic components of artificial neurons

In supervised learning, each input pattern/signal received from the environment associated with a specific desired target pattern. Usually, the weights are synthesized gradually, and at each step of the learning process they are updated so that the error between the network's output and a corresponding desired target is reduced. Back propagation (BP) algorithm will be introduced and applied as a supervised learning in this thesis.

On the other hand, unsupervised learning involves the clustering of, or the detection of similarities among, unlabeled patterns of a given training set. The idea here is to optimize (maximize or minimize) some criterion or performance function defined in terms of the output activity of the units in the network. Here, the weights and the outputs of the network are usually expected to converge to representations that capture the statistical regularities of the input data. Modified Hebbian algorithm will be introduced and applied as an unsupervised learning in this research.

Competitive learning can be categorized as a one of the unsupervised learning rule. However, the competitive learning rule can be used to train the weights in a competitive network, without knowing the prototype vectors. These categories are be discovered by the network on the basis of correlations in the input data. Thus, the network would classify each cluster of "similar" input data as a single output class. Learning Vector Quantization (LVQ) algorithm will be introduced and applied as a competitive learning in this research.

2. NETWORK STRUCTURE

2.1. Basic Assumptions

The major reason for applying neural network algorithm to statistical process control (SPC) in this research is to automate SPC chart interpretation and to develop knowledge based diagnosis for automotive body assembly. There are several basic assumptions to build effective pattern recognition system. They are as follows:

1) A network will be trained by only one pattern during a period time. It means that a trained network will only be able to identify a data set as being in one of defined nonrandom patterns.

2) To avoid ambiguity between patterns which can result poor convergence of the network, each nonrandom pattern must be generated as clearly as possible. For example, a small random noise-contaminated cyclic or systematic pattern can be classified as a natural pattern.

3) Since the main goal of this research is to improve type II error (or conclude the process is in control but actually is not) on a control chart, the most of training and testing data set will be generated within the control limit.

4) Knowledge and previous information about the manufacturing process, herein automotive body assembly process, to which this automatic pattern recognition system are already known to build an effective recognition system. For instance, it is impossible to construct effective pattern recognition system for identifying all sorts of cycle periods and amplitudes.

2.2. Training and Testing Dataset

Selection of the training data set is a key issue to the training of a neural network because it will strongly affect the performance of the networks. However, it is not easy to get such a training dataset, such as trend, cycle, mixture, and so on, from the process in practice. Thus, the pattern recognition algorithm proposed in this research is based on the assumption that the user has a set of patterns which is interested in detecting. The pattern generator designed to make a specified pattern has been described in Hwang and Hubele (1993). That pattern generator will be used to create both training and testing data sets in this research.

2.2.1. Natural/Random Pattern

First of all, natural pattern will be generated by general form which includes the process mean and random variations as follows.

$$y(t) = \mu + x(t) \tag{2.1}$$

Where $y(t)$ is measurement at time t and μ will be process mean when the process is in-control. The random noise, $x(t)$, will be expressed by random normal variate at time t, where $x(t)$ is $N(0, p\sigma_x)$ and σ_x represents process standard deviation when the process in-control. p will be the magnitude of random noise in terms of σ_x. The value of the p will vary between 0 and 1.

2.2.2. Upward/Downward Trends

The training and testing dataset for trends will be generated by following equation.

$$y(t) = \mu + x(t) + (t-t_\circ)m\sigma_x \tag{2.2}$$

Where m represents the slop of the trend in terms of σ_x. The value of the m will be positive for upward trends and negative for downward trends. The m between 0.2 and 0.4 by increasing 0.1 will be generated and trained the network for upward trends. Similarly, the m between -0.4 and -0.2 by increasing 0.1 will be generated the network for downward trends. If the slope m is more than 0.4 or less than -0.4, the dataset will be out of control limit. Thus, those cases are not considered in this research. The $t_\circ$ is a time reference point which indicates the starting point of this pattern.

2.2.3. Cycle

An equation for cyclic patterns with a disturbance component may be described as below.

$$y(t) = \mu + x(t) + \sin[2\pi(t-t_\circ)/T]k\sigma \tag{2.3}$$

Where k is the amplitude of the cycle in terms of σ_x, and $k > 1$. T is the period of the cycle. The selection of training and testing and testing patterns for cycle can be very complicated because it involves many parameters. To avoid ambiguity between patterns and unnecessary weight changes which can result in poor convergence, the period of cycle will be limited from 14 to 18 based on the process knowledge and past experiences. In this study, the period of 16 was selected to generate training and testing dataset.

2.2.4. Systematic Pattern

The systematic patterns will be generated by following equations.

$$y(t) = \mu + x(t) + (-1)^t Q\sigma_x \tag{2.4}$$

Where Q is the magnitude of the systematic pattern in terms of σ_x. The parameter values of Q must be less than 3.0, otherwise the most of data will be fallen out of control limit.

The equations from (2.2) to (2.4) in the pattern generator will generate several dataset by using several parameters that determine the specific shape of the pattern of interest. Each pattern defined as above must be generated as clearly as possible so that the pattern classes can be distinguishable as much as possible.

For example, an upward trend patterns with small slopes which are less than 0.1 may be classified as a natural pattern. The parameters for each pattern class should cover whole range of the defined domain as long as all points are within control limit. A subset of all possible values for these parameters will be investigated in this research. Details about training set are listed in Table 2.1.

At least 15 training dataset per each combination of parameters for unnatural pattern will be generated to train networks. Also, 100 dataset per each variation pattern may be randomly generated for testing the performance of the network.

2.3. Input and Output Layers

The input of network will be any specific value of the deviation from nominal position on each measurement point. Therefore, the number of input neuron must be equal to the number of observations a given time period in automotive assembly process. For both BP and LVQ algorithms, the input data will be same value as the deviation value. The general rule in ANN is that the input size should be as small as possible for efficient computation because the size of input usually determines the size and structure of the network. It is found that more than 16 input node does not improve significantly the performance of the network. Thus, 16 input nodes are selected in the first layer in the network.

Table 2.1. Parameter Values Used in the Training Set

Pattern	Parameter	Comment
Upward Trend	(0.1 : 0.05 : 0.4)	Slope
Downward Trend	(-0.4 : 0.05 : -0.1)	Slope
Cycle	(1.0 : 0.25 : 3.0)	Amplitude
Systematic	(1.0 : 0.25 : 3.0)	Amplitude

Note: (Initial value: Increment: Final value)

Table 2.2. Representation of the Output Categories

Pattern	Desired outputs				
	1	2	3	4	5
Natural	1	0	0	0	0
Upward Trend	0	1	0	0	0
Downward Trend	0	0	1	0	0
Cycle	0	0	0	1	0
Systematic	0	0	0	0	1

There are five output nodes corresponding to four unnatural patterns and a natural pattern of interest. The desired outputs will be used for both back propagation and LVQ algorithm.

2.4 Hidden layer and Transfer Function

There are no general rules to decide the number of hidden layers and the number of nodes in the hidden layer. In all cases found during this literature review, the number of hidden layers had been decided through trial and error. Guo and Dooley (1992) highlighted that fact there was no standard way of deciding the number of hidden layer and stated that as a rule of thumb either one or two hidden layers should be sufficient for almost any classification problem. To decide the number of nodes if the hidden layer, we can apply Kolmogorov's theorem that the maximum number of nodes in a hidden layer should be restricted to $2n+1$, where n is the number of input nodes. In this algorithm, the number of node will be less than 33(=2*16+1). Since too many nodes of hidden layer merely creates more chances for problems arising from local minimum, several cases of hidden layer will be performed and compared those performance to decide best fitted model in this research. Results showed the evolution of the network with different numbers of hidden nodes and 8 nodes in the hidden layer are appropriate in the network. Thus, 16-8-5 network gives best fitted model for nonrandom pattern recognition system in this research.

The back propagation algorithm works with any differentiable transfer function. The most widely used is the sigmoid (it is also called logistic sigmoid) function with output values ranging from 0 to 1. However, Guo and Dooley (1992), Hwarng and Huble (1993), Chang and Aw (1996) have encountered difficulty to detect directional invariance property using sigmoid function. Furthermore, since the output of the transfer function is used as a multiplier in the weight update equation, a range of output will be smaller when the summation is small and larger when summation is large [Cheng, 1997]. Another feasible transfer function is the hyperbolic tangent, with output values in the range from -1 to 1. The hyperbolic tangent function will provide equal weight to low and high end values. Thus, the hyperbolic tangent function will be used as a transfer function in this research. The LVQ algorithm will use the hyperbolic tangent function for hidden layer but simple binary transfer function will be used for output layer.

2.5 Train Network

During the training procedures, a learmines coefficient of 0.01 was used. The magnitude of this coefficient determines the pace of weight adaptation. Usually, a too large coefficient causes the convergence behavior to be oscillating and possibly never converge. On the other hand, a too small coefficient causes the learning process to progress slowly but has better chances to avoid local minimum. After several simulations of this network, the learning coefficient with a value 0.01 was the appropriate in this application.

The each pattern will be randomly and independently generated in the training set. Also, all patterns are equally represented in the training dataset to make performance comparisons among the pattern class. The weights for both BP and LVQ will be adjusted until the network converges to a pre-specified condition. The following steps are performed to train the network.

Step 1: Initializing Weight Vectors

At the beginning of training, the initial weights for both hidden layer and output layer are set randomly between -0.1 and +0.1.

Step 2: Presenting Input Vectors with Desired Vectors

Each input vector with the associated desired vector must be randomly selected and put in the network.

Step 3: Updating Weights

Modify weight matrixes by using Equation for BP algorithm and also modify weight matrixes for LVQ algorithm.

Step 4: Stopping Criterion

Repeat Step2-Step3 until all patterns are correctly classified or the required value of SSE has been reached for BP algorithm and the required iteration has been completed for LVQ algorithm.

2.6 Performance Evaluation

The performance evaluation was conducted to validate the usefulness of the proposed algorithms. Various combinations of parameters were used to test the classification ability of the network. Each testing data consists of 16 observations. To determine the on/off (1/0) state of the output node, any maximum value in the output vector set to 1 and other values set to 0. For example, if a real output vector is $[0.32, 0.01, -0.43, 0.03, 0.97]^T$, the classified output will be $[0, 0, 0, 0, 1]^T$ which indicates the systematic pattern. The performance of the Back propagation (BP) and Learning vector Quantization (LVQ) for each nonrandom pattern with various parameters is summarized from Table 2.3 to Table 2.10.

The performances on upward and downward trends for BP and LVQ are shown in the Table 2.3 to Table 2.6. As can be seen, both BP and LVQ algorithms have more than 98% accuracy to detect right classes. However, the performance rates with small random noise (e.g. = 0.1) are relatively lower than others.

The performance on cycles clearly depends on the amplitude of the cycle and the associated random noise. The results are summarized in Table 2.9 and Table 2.10. The performance to detect cyclic pattern is poor when the amplitude of the cycle and the associated random noise are relatively small. For instance, the accuracy of BP algorithm is 61% and 77% for cycles with (amplitude, random noise) = (1.5, 0.1) and (1.5, 0.2), respectively. On the other hand, the accuracy rate of LVQ algorithm for cycle is not too much sensitive to compare with BP algorithm as long the magnitude of cycle is large enough. This is obvious because a smaller random noise-contaminated cycle will be more likely natural pattern.

As can be seen in Table 2.9 and Table 2.10, some of systematic pattern with a small random noise can be classified as a natural pattern. Other cases, the performance of systematic pattern for LVQ algorithm is quiet consistent with various parameters. Whereas, the performance of systematic pattern for BP algorithm highly depends on the magnitude of random noise. Similar to cyclic pattern, it can be concluded that a smaller random noise-contaminated systematic pattern can be classified by a natural pattern.

From the overall results of performance evaluations, it can be concluded that LVQ algorithm has better capability for classifying nonrandom patterns than BP algorithm. At each learning iteration, the LVQ network is only told whether its input is correct or not and the neuron which wins the competition by being closest to the input vector is activated

Table 2.3. Performance Measurement of BP for upward Trends with Various Pattern Parameters

S : Slope SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
S=0.4 SD=0.1 (100)		100			
S=0.4 SD=0.2 (100)		100			
S=0.4 SD=0.3 (100)		100			
S=0.3 SD=0.1 (100)	1	99			
S=0.3 SD=0.2 (100)		100			
S=0.3 SD=0.3 (100)		100			
S=0.2 SD=0.1 (100)	7	93			
S=0.2 SD=0.2 (100)	2	98			
S=0.2 SD=0.3 (100)	1	99			

Table 2.4. Performance Measurement of LVQ for upward Trends with Various Pattern Parameters

S : Slope SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
S=0.4 SD=0.1 (100)		100			
S=0.4 SD=0.2 (100)		100			
S=0.4 SD=0.3 (100)		100			
S=0.3 SD=0.1 (100)		100			
S=0.3 SD=0.2 (100)		100			
S=0.3 SD=0.3 (100)		100			
S=0.2 SD=0.1 (100)	2	98			
S=0.2 SD=0.2 (100)		100			
S=0.2 SD=0.3 (100)		100			

Table 2.5. Performance Measurement of BP for Downward Trends with Various Pattern Parameters

S : Slope SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
S=0.4 SD=0.1 (100)			100		
S=0.4 SD=0.2 (100)			100		
S=0.4 SD=0.3 (100)			100		
S=0.3 SD=0.1 (100)	2		97	1	
S=0.3 SD=0.2 (100)	1		99		
S=0.3 SD=0.3 (100)			100		
S=0.2 SD=0.1 (100)	9		89	2	
S=0.2 SD=0.2 (100)	6		94		
S=0.2 SD=0.3 (100)			100		

Table 2.6. Performance Measurement of LVQ for Downward Trends with Various Pattern Parameters

S : Slope SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
S=0.4 SD=0.1 (100)			100		
S=0.4 SD=0.2 (100)			100		
S=0.4 SD=0.3 (100)			100		
S=0.3 SD=0.1 (100)	1		99		
S=0.3 SD=0.2 (100)			100		
S=0.3 SD=0.3 (100)			100		
S=0.2 SD=0.1 (100)	8		92		
S=0.2 SD=0.2 (100)			100		
S=0.2 SD=0.3 (100)			100		

Table 2.7. Performance Measurement of BP for Cyclic Pattern with Various Pattern Parameters

A : Magnitude SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
A=1.5 SD=0.1 (100)	39			61	
A=1.5 SD=0.2 (100)	23			77	
A=1.5 SD=0.3 (100)	8			92	
A=2.0 SD=0.1 (100)	6			94	
A=2.0 SD=0.2 (100)	1			99	
A=2.0 SD=0.3 (100)				100	
A=2.5 SD=0.1 (100)	2			98	
A=2.5 SD=0.2 (100)				100	
A=2.5 SD=0.3 (100)				100	

Table 2.8. Performance Measurement of LVQ for Cyclic Pattern with Various Pattern

Parameters

A : Magnitude SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
A=1.5 SD=0.1 (100)	22			78	
A=1.5 SD=0.2 (100)	2			98	
A=1.5 SD=0.3 (100)				100	
A=2.0 SD=0.1 (100)	1			99	
A=2.0 SD=0.2 (100)				100	
A=2.0 SD=0.3 (100)				100	
A=2.5 SD=0.1 (100)				100	
A=2.5 SD=0.2 (100)				100	
A=2.5 SD=0.3 (100)				100	

Table 2.9. Performance Measurement of BP for Systematic Pattern with Various Pattern Parameters

A : Magnitude SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
A=1.5 SD=0.1 (100)	23	12			65
A=1.5 SD=0.2 (100)		4			96
A=1.5 SD=0.3 (100)					100
A=2.0 SD=0.1 (100)	11	5			84
A=2.0 SD=0.2 (100)					100
A=2.0 SD=0.3 (100)					100
A=2.5 SD=0.1 (100)	1	1			98
A=2.5 SD=0.2 (100)					100
A=2.5 SD=0.3 (100)					100

Table 2.10. Performance Measurement of LVQ for Systematic Pattern with Various Pattern Parameters

A : Magnitude SD : Random Noise	Natural	Upward Trend	Downward Trend	Cycle	Systematic
A=1.5 SD=0.1 (100)	19				81
A=1.5 SD=0.2 (100)					100
A=1.5 SD=0.3 (100)					100
A=2.0 SD=0.1 (100)	3				96
A=2.0 SD=0.2 (100)					100
A=2.0 SD=0.3 (100)					100
A=2.5 SD=0.1 (100)					100
A=2.5 SD=0.2 (100)					100
A=2.5 SD=0.3 (100)					100

and allowed to modify its connection weight. This means that the LVQ algorithm is appropriate pattern recognizer when various parameters are involved in the system.

3. DIAGNOSIS OF THE NONRANDOM PATTERNS

After any nonrandom variations are detected or sustained on the automotive body, corrective action will be required in order to improve the dimensional quality. Roan (1993) and Ceglarek (1994) developed several case studies and strategies to locate the root causes for diagnosis of the process fault using knowledge of assembly process. However, they focused only on the sudden process change and large variations. Unfortunately, there are no any attempts to identify and diagnose regarding to nonrandom variation patterns in the assembly process. Based on their approach, diagnosis for nonrandom variation pattern defined if this research can be categorized by using manufacturing experiences, existing case studies and the knowledge of the assembly process. They are as follows:

1) Upward/downward Trends
   - Not enough spot welding
   - Tooling problems (clamp, welding)
   - Inconsistent dimensional quality of stamping process
2) Cyclic Pattern
   - Difference between measuring machine
   - Rotation of fixtures or gages
   - Regular movements of measurement sensing devices
   - Incorrect positioning for measuring machine station

3) Systematic Pattern
   - Difference in spread between different conveyors or shifts
   - Assembly fixtures
   - Locating holes
   - Worn positions or treads on locking devices
   - Loose holding arms

There are tremendous factors which can affect to dimensional quality. This means there are many possible root causes from any assembly stations, tools, fixture and etc. Above categorized root causes might be not enough to cover whole assembly process to help rapid corrective actions. However, it is a time-tolerated issue to build more detailed knowledge base for all possible causes of the dimensional variations. This issue will be left for the future working of this research. If any same variation patterns occur as does the old solved one, the root cause of the new case might be the same as that of the solved one. It will greatly narrow down to find possible causes and make it possible to rapid corrective reactions to improve dimensional quality. The following case study will demonstrate how the proposed network works for real data in automotive body assembly process in detail.

4. CASE STUDY

4.1 Data Collection

As a case study for neural control chart, the underbody assembly is presented. During body assembly operations, there are about 82 pallets which carry the auto bodies from station to station in the assembly line. Usually, there are 3 pin holes which support to position the underbody on the pallet. After underbody is loaded onto a pallet, the clamps of the pallet are closed to fix the position of the underbody on the pallet. Then, the underbody moves into each assembly station to weld subassemblies together. Clearly, underbody assembly process is very important because even small variations on the underbody will greatly affect to final BIW assembly. This section presents a case study frequently encountered in underbody assembly process. From the body shop assembly line, data was collected, n=419 per each measurement (total of 72 measurements) by the OCMM from the underbody assembly process.

4.2 Nonrandom Pattern Detection

To apply proposed algorithm, moving windows of data are presented to the network. The trained networks (BP and LVQ) tried to identify a nonrandom pattern based on the most recent 16 observations because, from the process control of view, the most recent data have important information for process control. If any nonrandom patterns are not detected, the network will try to identify a nonrandom pattern based on the second most recent 16 observations. For example, the first classification attempt is applied to observations, $\{x_{405}, x_{406}, x_{407, \ldots,} x_{418}\}$ and …… $\{x_{t-15}, x_{t-14}, x_{t-13}, \ldots, x_t\}$. Figure 1.1 shows the methodology of moving windows pictorially. As soon as data is preprocessed, each window of data is filtered through the trained network and determined one of predefined pattern classes (natural, upward trend, downward trend, cycle, or systematic).

All measured points were investigated by proposed network whether any nonrandom variation patterns occurred of not. As can be seen in Figure 4.1, systematic patterns were found on three points (UR2, UK1, UK2) around left rocker of the underbody~~.~~ Specifically, systematic patterns were detected on the UK2 from x_{150}, to x_{419}, and other are classified as natural. Similarly, UK1 has systematic pattern during $\{x_{80}, x_{81}, \ldots, x_{226}\}$ and $\{x_{272}, x_{273}, \ldots, x_{419}\}$. Also, systematic patterns were found on UK2 during $\{x_{143}, x_{144}, \ldots, x_{216}\}$ and $\{x_{258}, x_{259}, \ldots, x_{419}\}$.

From knowledge base for nonrandom variation pattern, systematic variation pattern can be from assembly fixtures or locating holes in the assembly process. Thus, this pattern suggested that the possibility of variation came from in part positioning. After investigation of the pallets, it was found that the three pin holes on the pallet had been worn out. The pin holes were replaced as a corrective action. After corrective action, new 216 samples were collected at the same measurement points and checked by the proposed network. The network classified the variation pattern for all three points are natural. Figure 4.1 shows run charts for same measurement points after corrective action. Therefore, we conclude that all measured points are in control.

The case study shows clearly how proposed neural network can detect and identify the predefined nonrandom variation pattern. Once these nonrandom patterns occur again on the run chart, the root causes of dimensional variations can be located systematically by investigating each possible cause based on the knowledge of the assembly process.

Therefore, it can be expected that the run chart with the proposed pattern recognition algorithm will play a more important role as a systematic diagnosis tool rather than only as a statistical monitoring tool.

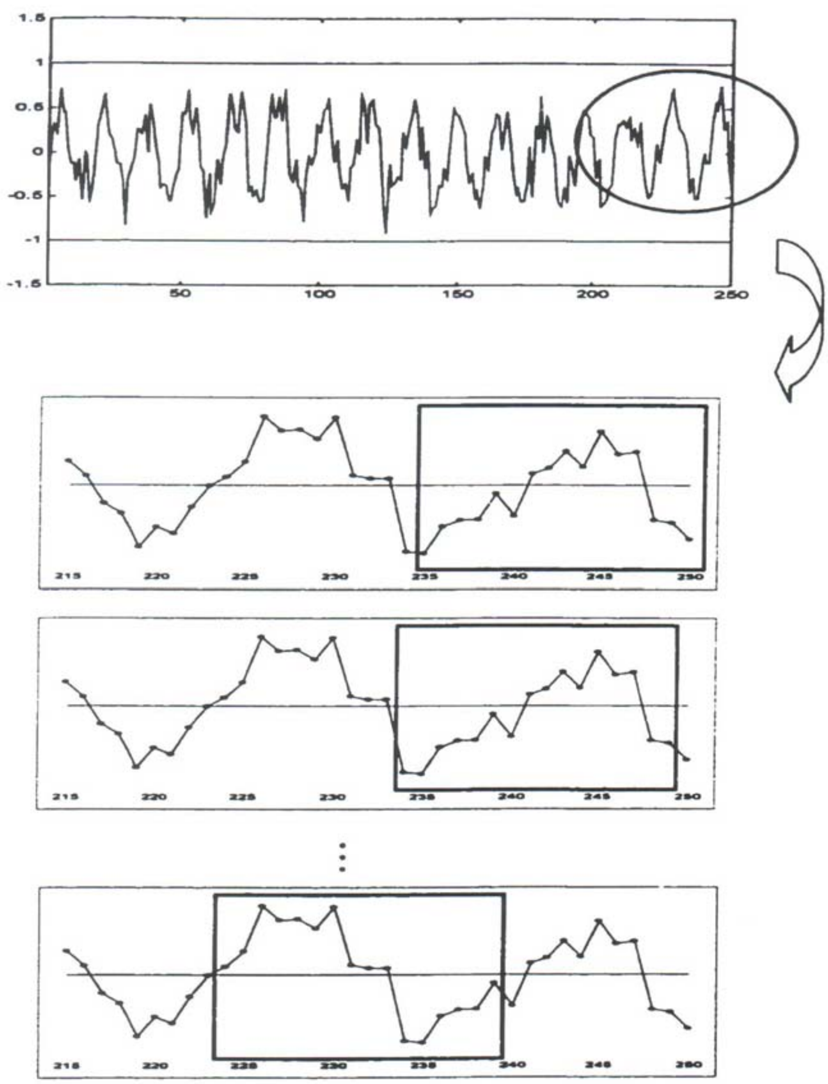

Figure 4.1. The concept of moving windows

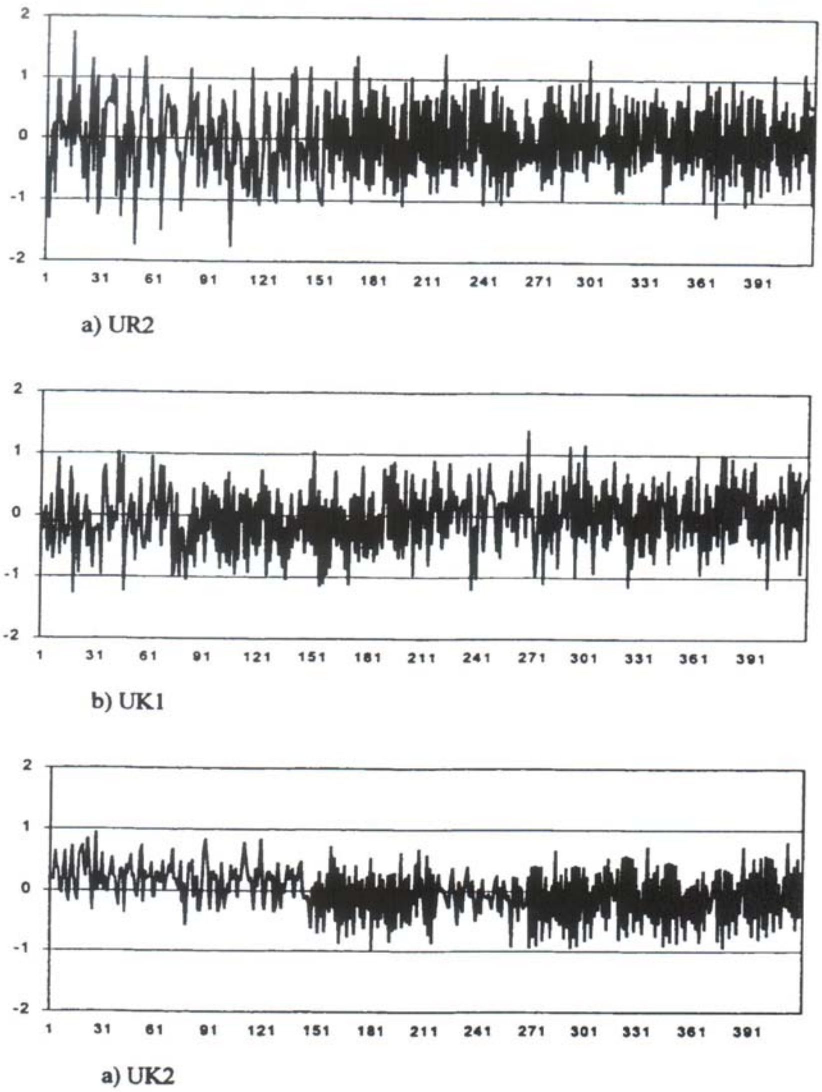

Figure 4.2. Run charts for three measurement points on the underbody

5. CONCLUSION

In this research, a control chart pattern recognition methodologies based on the Back propagation (BP) and Learning Vector Quantization (LVQ) algorithms were presented

Four nonrandom variations, which were upward trends, downward trends, cyclic pattern, and systematic pattern, were predefined to network. To train the networks, both BP and LVQ networks used the dataset generated by pattern generator with various combination of shape parameters and their performances were evaluated in terms of classification test.

An extensive evaluation indicated that LVQ performed better than BP. The proposed pattern recognition algorithm integrated with the process knowledge basis are designed not only to detect variation patterns, but also to address the identification of unacceptable variation manifested by nonrandom, or unnatural, patterns on the control chart. With this approach, the process can be monitored by a computer based pattern recognition algorithm without the need of human intervention. Accordingly, it contributes the reduction of an early failure rate of the item in the field.

REFERENCES

Baldi, P. and Hornik, K (1989) Neural network and principal component analysis: Learning from examples without local minima, *Neural Networks*, **2**, 53-58.

Baron, J. (1992) *Dimensional analysis and process control of body-in-white processes*, Ph.D. thesis, University of Michigan.

Ceglarek, D. (1994) *Knowledge-based diagnosis for automotive body assembly: methodology and implementation*, Ph.D. thesis, University of Michigan.

Chang, S. I. and Aw, C.A. (1996) A neural fuzzy control chart for detecting and classifying process mean shifts, *International J. of Production Research*, **34(8)**, 2265-2278.

Cheng, C.S. (1997) A neural network approach for the analysis of control chart pattern, *International J. of Production Research*, **35(3)**, 667-697.

Guo, Y. and Dooley, K. J. (1992) Identification of change structure in statistical process control, *International J. of Production Research*, **30(7)**, 1655-1669.

Han, W. (2002) A study on the experimental application of the artificial neural network for the process improvement, *J. of the Korea Society of Computer and Information*, **7(1)**, 174-183.

Hwang, H. B. and Hubele, N.F. (1993) Back-propagation pattern recognizers for control charts: Methodology and performance, *Computers and Industrial Engineering*, **24(2)**, 219-235.

Jang, K. Y. (2000) *Artificial Neural Network Approaches to Develop Robust Dimensional Data Analysis in Automotive Assembly Process*, Ph.D. thesis, Wayne State University.

Roan, C., Hu, S.J. and Wu, S.M. (1993) Computer aided identification of root cause of variation in automotive body assembly, *Manufacturing science and Engineering*, **64**, 391-400.