A non-conventional quality control system to detect surface faults in mechanical front seals

L. Barelli*, G. Bidini, F. Mariani, M. Svanziroli

Department of Industrial Engineering, University of Perugia, Via G. Duranti 1/A4, Perugia 06125, Italy

Received 14 September 2006; received in revised form 30 May 2007; accepted 25 November 2007

Available online 8 January 2008

Abstract

The Just in Time and the Total Quality policies have remarkably touched every field of modern industrial production. This context prompts companies to dedicate most of their efforts on researching and developing automatic systems of quality control to obtain the elevated standards of quality nowadays demanded by the market at every level of production. In fact the quantity of the exemplars allowed, which are not up to sample, is measured in parts per million in many sectors of production.

Many methodologies have been proposed to yield high quality in the industrial production lines, in order to provide surface examination and classification.

This paper describes an alternative system for surface analysis based on artificial neural networks (ANNs), developed in collaboration with the Italian manufacturer “Meccanotecnica Umbra S.p.A”. This system was implemented and tested in order to examine three particular surfaces of mechanical seals achieving good results in comparison with the deterministic system already implemented.

Keywords: Surface fault detection; Surface diagnosis; Artificial neural networks; Classification problem; Quality control system; Mechanical front seals

1. Introduction and methodology

This study aims to develop a quality control system to detect some particular faults, currently not recognized by the deterministic system already implemented by the Italian manufacturer “Meccanotecnica Umbra S.p.A”, that affect mechanical frontal seals. Consequently success relative to the detection of these faults means a direct improvement as compared with the methodology already implemented.

Therefore, the analysis is focused on the three particular faults currently not recognized and listed below:

- scratches on the bottom surface of the container (Fig. 1a);
- lack of sealing on the lateral surface of the container (Fig. 1b);
- faults on the ring carbon surface (Fig. 1c).

Considering how difficult it is to detect these faults and, on the other hand, the artificial neural network (ANN) ability in the classification problem in the case of complex and not linear phenomena, also in presence of disturbances and noise, the core of this system is based on “0–1 output” ANN. This peculiar architecture allows to train the ANN simply linking the pattern of perfect seals to output “0” and faulty seals to output “1”.

In the literature, different methodologies were developed for particular recognition and classification applications for process control, to guarantee a high-level quality in production processes, also by applying ANNs (Filho et al., 1999; Grunditz et al., 2004). A self-organizing feature map (SOFM) method, for example, divides image regions into classes (Jang Hee et al., 2001). The performance of the method is very high, also 100% in defect recognition.
through visual inspection of textile products (Tolba and Abu-Rezeq, 1997), while it slightly drops when extending the problem to classify to the metallic and wooden surfaces categories.

Moreover, *Neural network classifiers* to grade parts based on surface faults with spatial dependencies have been implemented in Schmoldt, 1994. This system is used in order to characterize and to classify faults on wooden surfaces.

Other methodologies relates to the *Markov random fields* (MRF) (Stachowiak et al., 2005), applied also for the classification of the oxidation level of metallic surfaces, assigning each pattern to the respective category according to the nature of the imperfections; while to analyze texture the *histograms and gray-level co-occurrence matrices* (GLCM) methods are used too (Manish et al., 2004; Palm, 2004; Baykut et al., 2000). Both MRF and GLCM methods yields a correct classification rate greater than 80%. Moreover, the nearest neighbor method is particularly suitable to problems of classification (Godin et al., 2004).

Therefore, the originality of this study is not so much in the application of ANN to classification problem for faults detection (or the development of learning algorithm modifications), but to the particular project of the solution, developed in agreement with phenomenon characteristics and constraints. In fact, system performances strictly depend on the parameters and indexes defined to characterize the phenomenon in reference to the faults presence or absence.

The system analyzes a restricted area of the mentioned surfaces, because the study just wants to demonstrate the reliability of the approach. Subsequently, an automated system for the mechanical frontal seals movement will be designed to guarantee the repetition of the analysis on all the zones of the component surface.

As concerns what discussed above, the first part of the paper illustrates how the digital images are treated and

<table>
<thead>
<tr>
<th>Nomenclature</th>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activation level</td>
<td>$a$</td>
<td>number of output nodes</td>
</tr>
<tr>
<td>Bias level</td>
<td>$b$</td>
<td>size of data vectors</td>
</tr>
<tr>
<td>Sigmoidal transfer function</td>
<td>$f$</td>
<td>weighted sum</td>
</tr>
<tr>
<td>Rotational speed</td>
<td>$N$</td>
<td>connection weight</td>
</tr>
<tr>
<td>Learning rate</td>
<td>$\eta$</td>
<td>node error</td>
</tr>
<tr>
<td>Sum weighted sum</td>
<td>$s$</td>
<td>sum</td>
</tr>
<tr>
<td>Connection weight</td>
<td>$w$</td>
<td>nodal error</td>
</tr>
<tr>
<td>Learning rate</td>
<td>$\eta$</td>
<td>nodal error</td>
</tr>
</tbody>
</table>

![Fig. 1](image.png)

Fig. 1. (a) Container fault: scratches on the bottom surface; (b) container fault: lack of sealing on the lateral surface; and (c) faults on the ring carbon surface.
what kind of information is deduced from defective elements along with good ones. Such an image processing procedure must supply the parameters then elaborated by the ANN to evaluate the surface conformity.

In order to carry out a supervised training on the basis of the experimental data deduced as detailed above, the authors have chosen the back propagation (BP) algorithm for training in consideration of its significant convergence performances in terms of both solution quality and reduced convergence time for the same error committed in the test. These characteristics have encouraged the authors to not execute a compared analysis of the performances concerning different training algorithms applied to the present case study.

Therefore, by applying multilayer feedforward ANNs trained with BP algorithm, different network architectures have been tested to find the best configuration in terms of performances and quality of the solutions produced. The results supplied by the ANN are compared with suitable threshold values, defined for each fault typology, which discriminate the cases of faulty seal and not-faulty seal.

The performances of the neural system have been finally verified also with noising images, in order to estimate the sensitivity of the system according to the quality of the acquired images. It must be noted that all the images used in the analysis were acquired by a common camera and not by means of a suitable vision system. Thus, also the quality of the original images is not very high and noise introduction was performed only for further sensitivity analysis of system performances.

2. Input parameters

As can be noticed from the images shown in Fig. 2, the surface area to analyze is dimensioned to be comparable with the dimensions of common faults, so that the surface imperfection meaningfully affects the statistical parameters extrapolated from such a zone.

Therefore, the images are acquired as RGB (1600 × 1200 pixels) and then converted to gray scale. On the basis of these data, it was noted that the presence of typical faults on the surfaces taken in consideration causes a sensible variation of the parameter calculated as the standard deviation of the pixels' brightness values. In fact, if the surface is good the values of pixels' brightness are very homogenous all over, while the dispersion of the values is enhanced when faults are present.

The neural system takes advantage from this disparity; in fact standard deviation of pixels' brightness was chosen as the parameter extrapolated from the images used as input for the ANNs. Moreover, in order to carry out a localized analysis, the zone under examination was divided in 25 sub-zones (20 × 20 pixels, Fig. 2) in consideration of its dimensions (100 × 100 pixels); so 25 sub-zones' standard deviation parameters must be calculated and form the input array of the system.

In relation to each fault, the ANN making up the neural system was trained using several images of good exemplars and images of elements that present some imperfections in the considered zone as well. The presence of imperfect exemplars in the training set is a necessary
condition to ensure a correct and complete training of the networks.

3. Container bottom surface analysis

In this case the training set collects 20 images of good exemplars and 20 images of seals with scratches on the container’s bottom surface. The unsuitable seals were placed in front of the camera so that the scratches were all along the analysis area. The testing set collects 20 images of good seals and 10 images of scratched ones. This set was used to test the performances of the trained neural system.

As already said, the input array collects 25 elements per image; consequently 25 neurons make up the outer layer of the neural network. The output layer collects only one neuron due to the fact that the network is forced to converge to “0” in case of good surface or “1” in case of scratches.

A lot of network structures were tested and the best grade of training convergence were reached in a structure characterized by three inner layers (see the scheme shown in Appendix A) with 50, 30 and 20 neurons, respectively.

The activation functions implemented in the input and inner layers are, respectively:

(1) tangent sigmoid,
(2) logarithmic sigmoid,
After 30 000 epochs and a calculation time of 15 min, the sum squared error (SSE) committed in the training phase was $2.147 \times 10^{-5}$, corresponding to a root mean squared (RMS) error of $1.015 \times 10^{-3}$. Both SSE and RMS parameters are calculated as indicated in Appendix A.

Fig. 3 shows the neural system performances in the case of the testing set input. As can be noted, the system output values are quite close to the real ones. In fact, only the first and the second outputs are not similar to the real ones and a value close to “1” is associated to all the scratched surfaces.

According to the results, it is possible to define a threshold equal to 0.2 to separate good exemplars from bad ones with 100% of faulty elements recognition.

4. Container lateral surface analysis

The system implemented to analyze the bottom metal surface of the container was adapted to control the quality of the lateral one. As already said, it is necessary to identify the lack of seal on this surface. Even in this case a training set and a testing one were collected.

In reference to the ANN architecture of Fig. A.1 in Appendix A, the best grade of training convergence was reached for the configuration 25-30-20-10-1. After 30 000 epochs and a training time of 15 min, SSE and RMS values equal to $2.55 \times 10^{-5}$ and 0.0011, respectively, arise.

Fig. 4 shows the implemented system performances. It is easy to note that, even in this case, the neural network is able to separate bad exemplars from the good ones and a threshold value of 0.3 is enough to yield a 100% correct classification.

5. Carbon surface analysis

Finally, the system was adapted to identify imperfections on the carbon surface of the rings that are the internal elements of mechanical seals. A training set of 40 (20 OK–20 NOK) images and a testing one constituted of 27 (18 OK–9 NOK) exemplars were collected.

In this case the configuration, characterized by a number of neurons in the input, inner and output layers of 25-30-30-10-1, guarantees the best grade of training convergence after 30 000 training epochs (convergence time of 15 min).
The training performances achieved are related to SSE and RMS values of $2.85 \times 10^{-5}$ and $5.4570e^{-004}$, respectively. The performances characterizing the neural system test are summarized in Fig. 5. The complete (100% recognition of faulty seals) separation between the good rings and the exemplars with imperfections is evident in consideration of a threshold value of 0.2.

6. System performance adding artificial noise to input images

As discussed in Section 1, all the images object of this study were acquired with a common camera and, therefore, already characterized by an intrinsic noise in comparison with the ones which in the implementation phase of the quality control system will be acquired by means of a suitable vision system.

Anyway, in order to carry out a sensitivity analysis of system according to the quality of the acquired images, it is interesting to verify the performances in the case of additional noise affecting the inputs.

To this aim, a random distribution of black and white pixels has been added to input images of the three testing sets. The system was tested with two levels of noise, a low and a high one. Figs. 6 and 7 show how both low and high level noises influence the images at the bottom surface of the seal container.

Moreover, Figs. 8, 9 and 10 show the outputs of the system in case of low and high level noise, respectively. It can be noted that in the case of low-level noise, the system is still able to separate scratched exemplars from the good ones in relation to the discriminant threshold value previously fixed at 0.2.

Moreover, a low-level noise, added to the seals images acquired with a common camera, does not affect the performances of the system even in the cases of lack of sealing and carbon rings analysis (threshold values of 0.3 and 0.2, respectively), as can be observed from the analysis of Figs. 11–13.

7. Conclusions

Applied to all classes of analyzed defects, the implemented method achieves high level of accuracy for not-high quality input images and also in the case of supplementary low-level noise introduction. In fact, as seen in the previous paragraphs, the neural classifier, applied to the three defects (lack of seal, scratches on the metal surface and carbon ring defects), is able to separate 100% of faulty...
elements from the good ones. One more important aspect to underline is that, once the system has been trained, you have only one parameter to set up: the threshold of separation. This means that little time is needed to suit the system according to user’s needs. In fact, if the user wants to change the selectivity of the system, only one parameter must be tuned; this characteristic makes this method truly user-friendly.

As the efficiency of the method is proved, the step to take is to re-edit the software in a different ambient of development (such as C++) and to optimize it. The goal is to reduce the processing time as much as possible to line it up to the production time and test the method directly on-line.

Appendix A

The general architecture of an ANN is shown in Fig. A.1. Fundamental elements are neurons (or cells), which form layers (input, output and hidden). Neurons in input and output layers represent the physical variables of the problem. The hidden layers neurons and the arcs, which connect the cells of all layers, represent activation coefficients and weights, respectively and are dynamically modified during the training phase.

The algorithm used during the training phase is the BP. The first activity of the training phase is the net initialization consisting of assigning a weight to each arc by using a random extraction algorithm, one of the most efficient of which is due to Nguyen and Widrow.

BP is one of the most widely supervised learning methods, in which an output error signal is fed back through the network changing the weights associated with each connection so as to minimize that error.

Generally, BP does not have to be fully interconnected, although most of the application work has been done with fully interconnected layers. In BP nets, the output value of each processing unit is called the “activation level” parameter. The BP learning algorithm involves a “forward-propagation step” followed by a “backward-propagation step”, which together represent a training loop (epoch).

The forward-propagation step is initiated when an input pattern is presented to the net. After the activation levels for the first layer of units is set, the next successive layer performs a forward-propagation step determining the activation levels of its following layer.

The forward-propagation step can be quantified by looking at Fig. A.2, where a typical processing unit for an ANN is shown. A processing unit has many input signals, each arriving from another unit, but only a single output signal. To compute its output, each unit applies to its inputs a “transfer function”\( f \), also defined as non-linear threshold function.

The first operation performed in the processing unit is a weighted sum of the activation levels\( a_i \) sent by each node\( i \) of the preceding layer, according to the relation

\[
\text{sum}_j = b_j + \sum_i a_i w_{ji}, \tag{A.1}
\]

where \( w_{ji} \) is the weight of the connection which links unit\( i \) to unit\( j \) and \( b_j \) is the bias value.

In the second step the transfer function \( f \) is applied, using a sigmoid function the general form of which is

\[
f(\text{sum}_j) = \frac{1}{1 + e^{-\text{sum}_j}}. \tag{A.2}
\]

The resulting value becomes the new activation level of unit \( j \). This output value is sent along all the output interconnections. In the scheme of Fig. A.1 the typologies
of the transfer functions implemented in the ANN architectures tested are indicated for the neurons input and hidden layers.

The error correction step takes place after the forward-propagation step is completed. Each output layer node produces a real number which is compared to the "target output" specified in the training set. In the backward-propagation step the weights are adjusted for all interconnections. For the output layer units the error value \( \delta_j \) is
\[
\delta_j = (t_j - a_j)f'(\text{sum}_j),
\]
where \( f'(\text{sum}_j) \) is the derivative of the sigmoid \( f \), \( a_j \) the output value for unit \( j \), \( t_j \) the target value for unit \( j \).

The error values of the units in the other layers (Fig. A.3) are expressed by
\[
\delta_j = \left( \sum_k \delta_kw_{kj} \right)f'(\text{sum}_j).
\]
Each interconnection weight is then corrected according to the weight adjustment defined by the relation of Rumelhart and McClelland
\[
\Delta w_{kj} = \eta \delta_k a_j,
\]
where \( \eta \) is the "learning rate", a real number commonly falling into the interval 0.2–0.8. Its value, determined by a suitable sub-model is varied dynamically, since large \( \eta \) values favor the ANN convergence speed, but can also lead to instabilities in the convergence itself. The forward/backward activity stops when a target value for the SSE is reached: the ANN is then considered to be trained. The SSE value can be expressed by
\[
\text{SSE} = \sum_{p=1}^{n_r} \sum_{o=1}^{n_o} (t_{op} - x_{op})^2,
\]
where \( (t_{op} - x_{op}) \) is the difference between the target experimental output and the net output, \( n_o \) is the number of output nodes, \( n_n \) is the size of input and output vectors.

Usually the ANN performance is measured by the RMS error value:
\[
\text{RMS} = \sqrt{\frac{\text{SSE}}{n_r n_o}}.
\]

It should be remarked that the pattern set and the target set data must be concurrently measured, i.e. the data must establish a one-to-one correspondence in time:
\[
g(t) = F[x(t), y(t), z(t)].
\]

References
