Neural Network Classification of Diesel Spray Images

S. D. Walters, S. H. Lee, C. Crua and R. J. Howlett

Engineering Research Centre, School of Engineering
The University of Brighton, Brighton, U.K.
s.d.walters@brighton.ac.uk
http://www.brighton.ac.uk

Abstract. This paper describes an evaluation of a neural network technique for modelling fuel spray penetration in the cylinder of a diesel internal combustion engine. The model was implemented using a multi-layer perceptron neural network. Two engine operating parameters were used as inputs to the model, namely injection pressure and in-cylinder pressure. Spray penetration length were modelled on the basis of these two inputs. The model was validated using test data that had not been used during training, and it was shown that semi-automated classification of complex diesel spray data is possible. The work lays the foundations for the establishment of an improved neural network paradigm for totally automatic, fast, accurate analysis of such complex data, thus saving many man-hours of tedious manual data analysis.

1. Introduction

In recent years, research has shown that the combustion and emission characteristics of modern diesel engines are dependent on a multitude of parameters, including: fuel atomisation, nozzle geometry, injection pressure, shape of inlet port, and other factors. In order to improve air-fuel mixing and the resulting combustion, it is important to understand the underlying fuel atomisation and spray formation processes. Experimental and theoretical approaches have been adopted by researchers in order to investigate the characteristics of spray behaviour, formation and structure for the high-pressure injector, in order to improve engine efficiency; in practice the goals are: improved combustion, performance and reduced exhaust emissions. However, further detailed studies of the atomisation characteristics and spray development processes of high-pressure diesel sprays are still needed.

Intelligent systems, i.e. software systems incorporating artificial intelligence (AI), have demonstrable benefits for control and modelling of engineering systems. For example, they feature the highly useful ability to rapidly model and learn characteristics of multi-variate complex systems, offering performance advantages over more conventional mathematical techniques. This has resulted in their application in diverse applications within power systems, manufacturing, optimisation, medicine, signal processing, control, robotics, and social/psychological sciences [1-3].

The AI approach is well-suited to the analysis of many combustion-related problems; AI has considerable potential for making faster, more accurate predictions than some traditional methods. Increasingly, the availability of complex sensory and computing systems is resulting in the production of vast quantities of information-rich data. Historically, the analysis of such complex data has required very substantial human effort, often with every image of every video needing to be evaluated by human eye. The aim of this investigation, and the partner investigation featured in [4, 5], has been to apply intelligent systems tools and techniques to the problem of effectively, and semi-automatically, processing and analyzing large complex data sets. The data in question was generated during advanced experimental engine research.

The specific objectives of the experimental work described in this paper were to:

1. Create a semi-automatic diesel spray analysis system in software;
2. Investigate the effect of altering key system parameters, namely:
   - Number of hidden nodes employed, and,
   - Desired output layer threshold;
3. Evaluate the system and assess its future potential.

2. Acquisition of Diesel Spray Penetration Video Clips

2.1 Image Data

A large repository of complex data was produced in the course of an extended investigation using a Ricardo Proteus research engine. These data comprised of '.AVI' video clips depicting the developing spray patterns of diesel injection processes, under selected operating conditions of: fuel injection pressure, in-cylinder air pressure, and injector nozzle size and type [6]. EXCEL files containing additional information about the engine operating conditions were associated with the video clip files, thereby providing a complete set of results for analysis.

It was desired to have the ability to classify images from the videos, using the properties of the images and according to the prevailing engine conditions. In order to classify the images from the video clips, the resulting software program needed to handle vast volumes of data, undertaking image processing, pre-processing and, in future work, subjecting it to neural network analysis by an internal neural network algorithm, before finally the outputting the results. To initially control complexity of the system, it was decided to use a custom-written multi-layer perceptron (MLP) neural network, implemented in the C language [7, 8], as the analysis engine. All other sections of the system were included in specially-written front-end MATLAB ® V.7 based software [9, 10].

The images representing the time-varying spray process under sets of injection and in-cylinder pressure conditions were processed using a specially-written program, coded in MATLAB. The use of MATLAB allowed the data to be relatively quickly visualised and analyzed without the need for writing too many low-level language routines, for example to control file access, perform image processing tasks and display outputs such as graphs. At a convenient juncture in the future, the MATLAB front end could be converted to a C-language addition to the MLP software, or the neural network could be implemented in MATLAB after the other routines, according to the needs of the application.

2.2 Image Processing

In recent years, the field of image processing has advanced apace; MATLAB features an optional Image Processing Toolbox which proved efficacious and convenient [10]. The actual methods chosen were derived from studying the properties of the spray-cone images, coupled with extensive reference to the literature [11-13]. The potentially high degree of spray break-up and the desire to model the spray in a transparent fashion favoured a relatively simple encoding method that consistently produced a fixed data word-length, as opposed to the relatively homogeneous spray shown as an example by Batchelor et al. [14].

The data processing technique was as follows. Each of the 80 .AVI video clips comprised 300 images, each of 320x128 pixels. Key images were automatically selected from each video – in this case those images in which peak spray penetration was attained; Fig. 1 shows examples of the peak penetration images for each of the four classes of data.

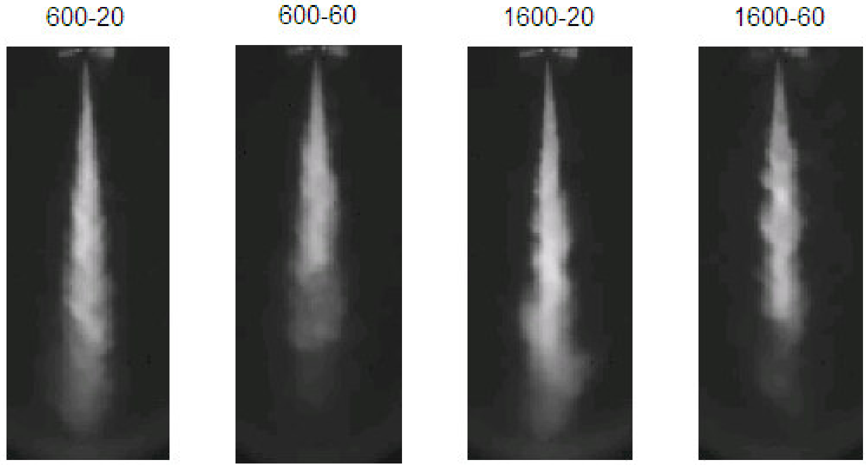

Fig. 1 Peak Penetration Spray Image Examples for each Set of Conditions

The grey-scale images were thresholded [10], yielding binary (black and white) images, as shown in Fig. 2.

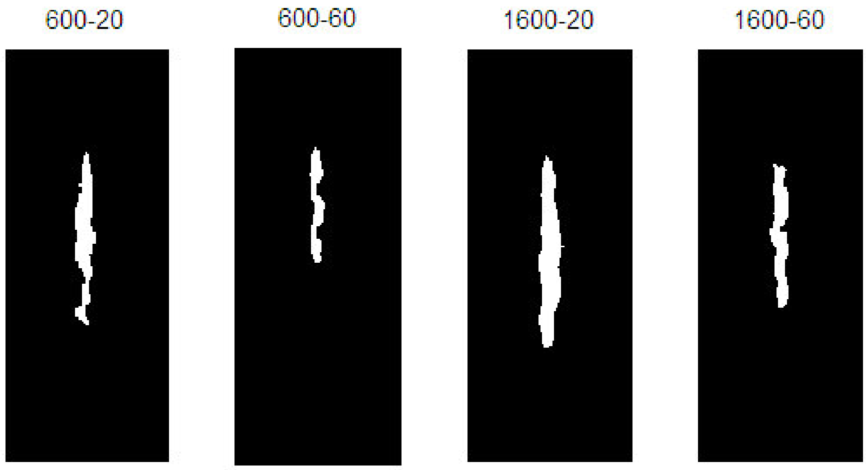

Fig. 2 Thresholded Spray Data for each Set of Conditions

A first pass through all of the videos facilitated the automatic selection of the optimum scanning window size to reduce redundant unchanging areas of black pixels, and identified the location and data of the 'peak penetration image' in each video. On the second pass through the data, the desired 80 peak images were selected out and re-sampled accordingly, reducing the window size to 32x7 pixels, hence significantly reducing the number of data-points referred to the neural network. Figs. 3 and 4 indicate the level of detail discernable after re-sampling. The reduced images were scanned line-by-line, producing a total of 80 vectors, each of which featured 224 data-points.

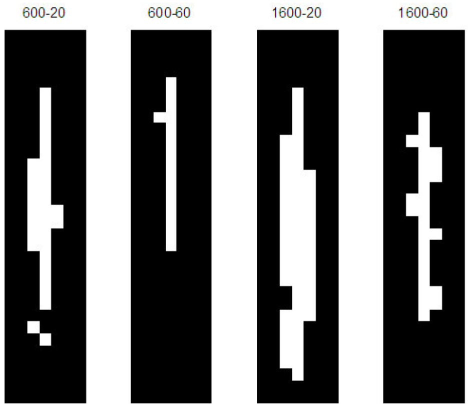

Fig. 3 Re-sampled Spray Data for each Set of Conditions

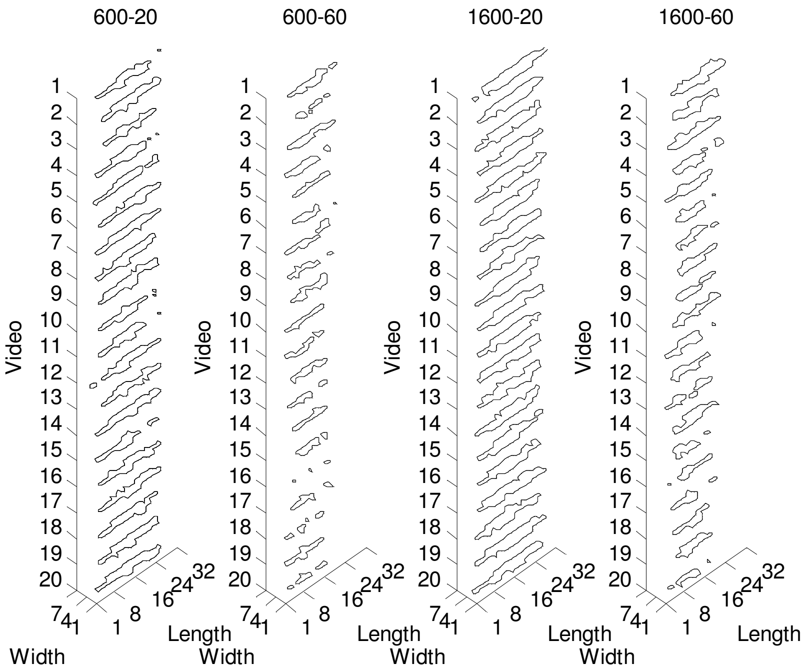

Fig. 4 Visualisation of all the Training (1 – 10) and Recall (11 – 20) Data

Fig. 5 depicts as vectors the same image examples as shown in previous figures Figs. 1-3. Finally, these vectors were combined with suitable 'Desired Output' information, yielding two data files for evaluation of the neural network in 'Training' and 'Auto-recall' modes. 'Auto-recall' is an automated routine, in which a set of previously unseen data exemplars are subjected to a 'Recall' process; the results yield network performance statistics.

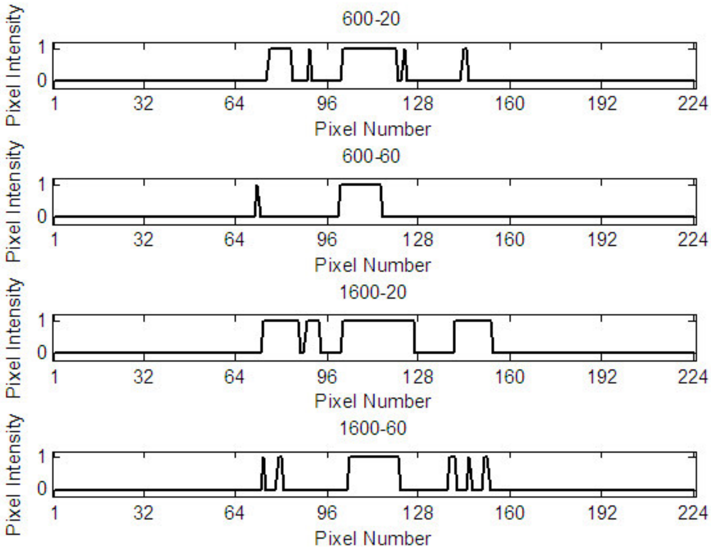

Fig. 5 Examples of Neural Network Input Data for each Set of Conditions

3. Effect of Varying the Number of Hidden Nodes

3.1 Method

Two specially-written MATLAB m-files were run in sequence. The first program calculated the minimum image area suitable for processing any choice of images from the videos by neural network; the second used this information to process the images and prepare the training and recall data.

The resulting training and recall files were passed to NDA (Neural Data Analyzer, V.7.0), running on the same PC. The NDA software package is an MLP neural network, implemented in the C-language, in-house at the University of Brighton.

After starting NDA, network architecture and training parameters were entered, on the Network Set-up and Training screens, respectively; these parameters are shown in Table 1. The neural network was trained using the training file and a range of different numbers of Hidden Nodes, as indicated in Table 1. In order to retain a degree of detail in the images, thus indicating spray break-up and overall shape, 224 input data points, hence input nodes, were chosen, as described in Section 2.2. According to the results shown later, this necessitated a fairly large number of hidden nodes, given training data of just 80 exemplars, despite the prediction of a training heuristic [8]: 'For good generalisation, the condition: $n_w <=\sim n_t <=\sim 10n_w$ should be satisfied, where n_w = Number of network weight values and n_t = Number of training examples'.

Table 1. Neural Network Configuration and Operating Parameters

Parameter	*Value*		*Parameter*	*Value*
Input Nodes	224		H Layer Learn. Rate	0.080
Hidden Nodes	10/25/38/50/63/75		O Layer Learn. Rate	0.020
Output Nodes	4		Momentum	0.800
Training Sets	40		Limit	0.200
O Layer Act. Func.	Sigmoid		Initial Value	0.500
O Layer Threshold	0.1		Error Interval	10

3.2 Results and Discussion

Training was accomplished quickly, in the worst case taking just over 6 seconds. Three training sessions were undertaken for each set of operating conditions, and then the mean was taken of each trio of results. Peak 'Training Algorithm Iterations' and 'Solution Time', and reduced 'Training Error' were to be found at 50 – 63 hidden nodes, although it should be noted that generalisation performance may not necessarily have been optimal at the lowest error point.

Mean Recall results ('Correct Outputs', 'Correct Output Sets' and 'Matching Greatest Output') are shown in Fig. 6. 'Correct Outputs' refers to the total percentage of neural network output values that registered the desired output, +/- an 'Output Layer Threshold' value; in this first experiment the Threshold was set at 0.1 (i.e. +/- 10% variation allowed). 'Correct Output Sets' refers to the percentage of output sets in which the outputs were all within the 'Output Layer Threshold' tolerance of the desired outputs. 'Matching Greatest Output' refers to the percentage of output sets in which the neural network correctly identified the 'winning' output class by setting its output node to the greatest value within the set.

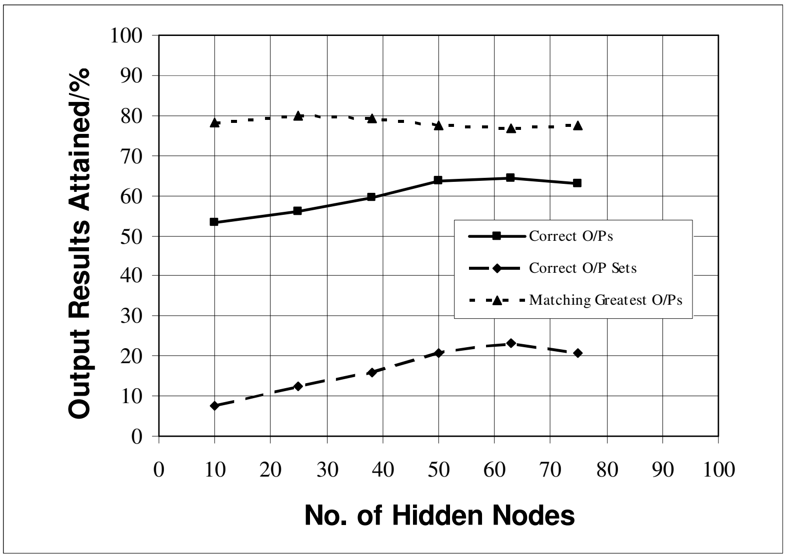

Fig. 6 Variation of Neural Network Performance with Number of Hidden Nodes

The network was best able to maintain +/-10% accuracy of ALL its outputs over 65% of the unseen recall file data, when the number of hidden nodes was between 50 and 75; this is shown by the 'Correct Outputs' plot.
Relatively poor performance was achieved on producing exactly 'Correct Output Sets' that were all within +-10%; the best performance was found to be 23% for 63 hidden nodes. Since the neural network was only being expected to choose the winning output node in this test, this poor performance was not considered very important. In an industrial environment, there would probably have been a different output configuration, for example, two output nodes with linear activation functions, each representing an engine parameter. In addition, the chosen 'Output Layer Threshold' value of 0.1 (or +-10%) was considered to be demanding for this initial experiment.
The 'Matching Greatest Outputs' graph plot was considered important for this experiment, as the network was expected just to choose the greatest output as the correct one. Best performance was 80%, attained between 25 and 38 hidden nodes.

3.3 Conclusions

The MLP neural network was trained quickly and effectively for all values of 10 – 75 hidden nodes.
An output threshold value of 0.1 (or +-10%) was considered strict for this experiment.
The choice of output configuration i.e. four outputs for four classes was not ideal for this application – two linear nodes would be more appropriate and may give better results.
Based on the aforementioned results, the best overall performance was considered to be achieved between 50 and 63 hidden nodes, so the minimum number to give best performance was chosen for future work, i.e.: 50.

4. Effect of Varying the Output Threshold

4.1 Method

In accordance with the results of Section 3, the configuration of 50 hidden nodes was retained for future work. The three training sessions with 50 hidden nodes had previously enabled the production of three sets of weights which could be used for recall using different Threshold values, without retraining the neural network again.
For this experiment, the neural network configuration and recall parameters were as shown in Table 2. The appropriate weights files were loaded into NDA, and recall tests were performed for four different Threshold values.

Table 2. Neural Network Configuration and Operating Parameters

Parameter	***Value***		***Parameter***	***Value***
Input Nodes	224		H Layer Learn. Rate	0.080
Hidden Nodes	50		O Layer Learn. Rate	0.020
Output Nodes	4		Momentum	0.800
Training Sets	40		Limit	0.200
O Layer Act. Func.	Sigmoid		Initial Value	0.500
O Layer Threshold	0.1/0.2/0.3/0.4		Error Interval	10

4.2 Results and Discussion

Fig. 7 shows the results obtained with the various ‘Output Layer Threshold’ values, ranging from 0.1, as used in the previous experiment, through 0.2 and 0.3, to the much less- demanding value of 0.4.

It may be readily seen that the number of overall ‘Correct Outputs’ increased from 64% to 87% as the ‘Output Layer Threshold’ was increased to 0.4.

‘Correct Output Sets’ showed a large increase in performance from 23% to 68%.

‘Matching Greatest Outputs’ remained constant at 78% throughout the experiment, showing a robust response regardless of threshold value.

This work relied on the principle of ‘Winner-takes-all’, i.e. the output node exhibiting the greatest value was the winning node in the output set. For this application, the neural network appeared to be relatively undemanding as to the Number of Hidden Nodes and ‘Output Layer Threshold’ adopted for training and recall.

Actual output values from the network may attain greater importance in future variations of this work, and methods for increasing the overall performance are currently subject to experimentation. For example, improvements by: (1) removing redundant white area in the image area, in addition to black; (2) re-sampling with even better choice of pixel reduction filtering; and (3) optimizing the number of hidden nodes to the nearest node, rather than to within 10-15 nodes.

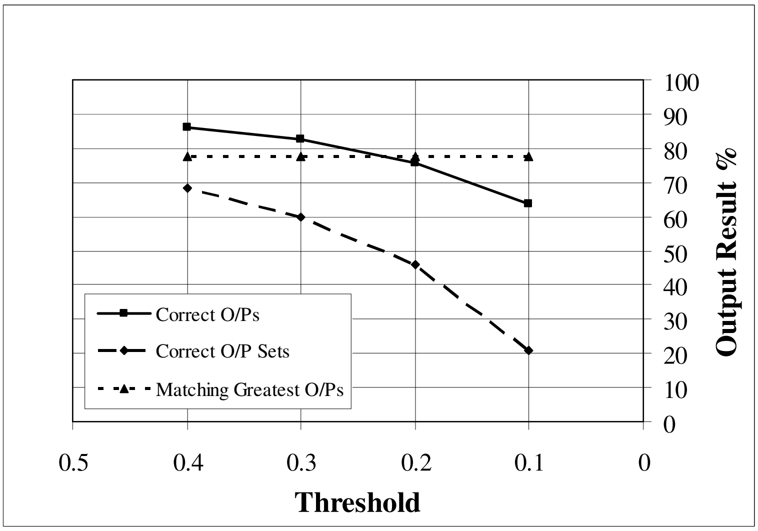

Fig. 7 Variation of Neural Network Performance with Changing Output Threshold

4.3 Conclusions

Given less-demanding ‘Output Layer Threshold’ values, there was an apparent increase in the performance of the neural network. However, for “winner-takes-all” testing, i.e. using the ‘Matching Greatest Outputs’ value, the results were exactly the same, at 78%, throughout all the tests.

For selection of classes of diesel spray, the neural network test was robust, but could be refined further, given more time; recommendations have been made.

Automatic image processing needs to incorporate safeguards against corrupted images in large data sets.

5. Further Work

The performance of the devised neural network solution was considered satisfactory for an initial experiment. However, future work includes:

- Refinement of the image processing techniques. This will ensure that unchanging pixels (confusing to the neural network) are minimized, assisting with the following analysis.
- Experimentation with the precise number of hidden nodes required.
- Future engine tests to capture more engine parameters to enable more complete modelling of the spray process and evaluation of analysis methods.
- Adoption of new neural network algorithms for elimination of outlying data points and increased processing speed is under consideration.
- Neural network training theory indicates that the amount of training data used for this work would appear to be considerably less than optimal. However, the results obtained, using previously unseen test data, indicate that the technique has considerable merit for further investigation using larger training and recall data sets.

Acknowledgements

- This work has been funded by the European Union under the Interreg III Programme of the European Regional Development Fund, Project name: 'Intelligent Engine II', Project Number: 126.
- The authors gratefully acknowledge the contributions of Dr. D. H. Lawrence, Dr. Steve Begg, EPSRC Loans Pool and Ricardo Consulting Engineers Ltd in providing advice, data and equipment for this paper, respectively.

References

1. Kalogirou S.A.: Applications of artificial neural-networks for energy systems, Applied Energy 67, 2000, pp.17–35.
2. Xu K., Luxmoore A.R., Jones L.M., Deravi F., Integration of neural networks and expert systems for microscopic wear particle analysis, Knowledge-Based Systems 11, 1998, pp.213–227.
3. Walters, S. D., Howson, P. A., Howlett, R. J.: Production Testing of Spark Plugs using a Neural Network, Paper No. kes05-159, Vol. 4, pp. 74-80, Proceedings, KES2005, Melbourne, 14th-16th September 2005, ISSN 0302-9743, ISBN 3-540-28897-X.
4. Lee, S. H., Walters, S. D., Howlett, R. J.: An Adaptive Neuro-fuzzy Modelling of Diesel Spray Penetration, Paper No. SAE-NA 2005-24-64, Proceedings, ICE2005, Capri, 11th-16th September 2005, ISBN 88-900399-2-2.
5. Lee, S. H., Howlett, R. J., Walters, S. D., Crua, C.: Fuzzy Logic and Neuro-fuzzy Modelling of Diesel Spray Penetration, Paper No. kes05-388, Vol. 2, pp. 642-650, Proceedings, KES2005, Melbourne, 14th-16th September 2005, ISSN 0302-9743, ISBN 3-540-28895-3.

6. Crua, C.: Combustion Processes in a Diesel Engine, Ph.D Thesis, University of Brighton, 2002, http://www.crua.net/thesis .
7. Walters, S. D.: Characterization and Analysis of Kettering-type Automotive Ignition Systems and Electrical Spark Profiles, Ph.D Thesis, University of Brighton, in Association with Champion Spark Plug (Europe), 1998, pp. 35-36.
8. Haykin, S.: Neural Networks – A Comprehensive Foundation, 2nd Edition, Chapter 4, pp. 156-255, ISBN 0-13-273350-1, Prentice Hall, Pearson Education, 1999.
9. MATLAB Manual (version 6, Hardcopy), August 2002.
10. MATLAB Manual (version 7.1, Online), August 2005.
11. Forsyth, D. A., Ponce, J.: Computer Vision – A Modern Approach, ISBN 0-13-085198-1, Prentice-Hall, 2003, Ch. 8 pp. 165-189.
12. Rosandich, R. G.: Intelligent Visual Inspection using Artificial Neural Networks, ISBN 0 412 70800 0, Chapman & Hall, 1997, pp. 116-136.
13. Batchelor, B. G., Hill, D. A., Hodgson, D. C.: Automated Visual Inspection, North-Holland (ISBN 0-444-87577-8) / IFS Publications Ltd. (ISBN 0-903608-68-5), 1985, pp. 329-398 and 535-547.
14. Batchelor, B. G., Hill, D. A., Hodgson, D. C.: Automated Visual Inspection, North-Holland (ISBN 0-444-87577-8) / IFS Publications Ltd. (ISBN 0-903608-68-5), 1985, Case Study 22, pp. 501.
15. Lippman, R. P.: An Introduction to Computing with Neural Networks, IEEE ASSP Magazine, Vol. 4, pp. 4-22, 1987.
16. Thompson, S., Fueten, F., Bockus D.: Mineral Identification using Artificial Neural Networks and the Rotating Polarizer Stage, Computers and Geosciences, Vol. 27, pp 1081-1089, 2001.
17. Howlett, R. J., De Zoysa, M. M., Walters S. D.: Monitoring IC Engines using Neural Networks, Paper: SAE-NA 2003-01-09, Procs., 6th International Conference on Engines for Automobiles, Capri, 2003, On CD.
18. Hush, D. R., and Horne, B. G.: Progress in Supervised Neural Networks, IEEE Signal Processing Magazine, pp. 8-39, 1993.
19. Themistoklis, K.: Implementation of an Automated Fuel Spray Image Analysis System, BEng Final Year Report, University of Brighton, 10/5/2001, Ch. 1-2, pp. 2-14.
20. Schalkoff, R.: Pattern Recognition, Statistical, Structural and Neural Approaches, ISBN 0-471-55238-0, J. Wiley & Sons, 1992, Ch. 10, pp. 204-220, and Ch. 12, pp. 236-259.
21. MATLAB Neural Network Toolbox Users' Guide, (version 4, Hardcopy), March 2001.