Applying Neural Networks to Fault Classification in Image Data

Nicholas D. Freear, James M. Shippen and Ronald L. Westra*
The School of Manufacturing and Mechanical Engineering
University of Birmingham, Birmingham, B15 2TT
The United Kingdom
Phone: +44-121-414-4185, Fax: +44-121-414-3958
email: {n.d.freear, j,m,shippen}@bham.ac.uk
*The Department of Mathematics
Maastricht University
The Netherlands
Phone: +31-43-388-3722, Fax: +31-43-321-1889
email: westra@math.unimaas.nl

ABSTRACT: This paper presents a comparison of linear correlation and neural network algorithms for classifying and correcting printing faults in real-time. Following image acquisition and pre-processing, pixel intensity profiles are extracted across known error sites. The linear correlation approach classifies these features against a small number of profile models giving image invariance, then against the large set of visual faults. The first neural network replaces only the first mapping, and has outperformed the linear algorithm. The second feed-forward network is substituted for the final linear correlation, and has shown limited success. Further tests are being performed with real print samples, and a hybrid of linear and adaptive routines may be applicable to other applications.

KEYWORDS: Multi-layer perceptron, feed forward, back propagation, machine vision, control, printing.

INTRODUCTION

Commonly in industrial processes, including pad printing operational pressure dictates that human input is reduced, the production rate is increased and the occurrence of faults is reduced, whenever possible. Pad printing is the only commercial system to doubly curved surfaces, and it is used to decorate many consumer products where quality is especially important. Therefore, a real-time autonomous visual inspection and control system is being developed, that will classify visual defects, and combine the outcome with sensor data and *a priori* knowledge to make corrective actions. This paper presents a comparison of linear correlation (LCA), and neural network (NN) classification algorithms, with the bench-mark linear correlation explained first. In the pad printing process ink is picked up from a flat etched metal plate by a deformable rubber pad and deposited on the product.

PATTERN RECOGNITION AND CONTROL

Pattern recognition requires that characteristic features be extracted so that they may be classified, and as an image contains a lot of data extraction is a significant step. An image of the suspect graphic is acquired then normalised with respect to geometry, and illumination. If the correlation with the template image is below a threshold the graphic contains visual defects, and the suspect image is matched with the template. Allowing for noise and distortion, the match or absolute difference reveals features which can be extracted from the image as scan profiles (1-dimensional) or surfaces (2-dimensional) that contain the fault. These are vectors and matrices respectively containing contiguous pixel values or intensities, and initially 1-dimensional grey-level data has been employed. Hue and saturation data may also be employed.

Eighteen printing faults and a '*good*' graphic have been identified in the pad printing process, k=1…19; these include *ink missing*, *air bubbles*, and ink filaments stretched from the printed boundary. An example of the latter fault, often referred to as *hairs*, is shown in figure 1a, and it can be caused by high ink viscosity or a static electricity charge on the printing pad.

The approach used to classify these faults involved the fitting of scan lines across the captured image. Five curve types or positions were defined relative the 'good' graphic in the template, and they are termed Γi, i=1…5. Γ1 crosses the printed boundary along the local normal; Γ2 is positioned just inside the printed area parallel to the printed boundary; Γ3 is parallel to and just outside the printed boundary; Γ4 is positioned deep inside the printed area; and Γ5 is distant from the printed area. Hairs, for example, are best detected by examining Γ3. There are two ideal expected outcomes for these profiles *E*1, a straight line or constant intensity, and *E*2, a step. In the template a profile in position Γ1 leads to *E*2, while Γ2…5 gives the expected *E*1. The actual profile, *A*i, must fit one of a finite number of models or match types, *R*j, j=1…6 for a single fault in the linear correlation algorithm. For instance, *R*1 is 'OK' which is *E*1 with noise, *R*2 is 'too low', and *R*6 is 'oscillation'. Combinations of faults increase the possible outcomes to thirty, j=1…30.

LINEAR CORRELATION ALGORITHM

The Linear Correlation Algorithm (LCA) employs one-dimensional discrete convolutions between the actual profile *A*, and the set of match types *R*j, compared with corresponding convolutions between the expected outcome *E* and *R*j. The resulting vector, match *M*ij, is image invariant. Linear cross correlation of the match against a support matrix, *S*ijk, gives a probability *P*ijk for each fault. Probabilities from multiple scan lines are accumulated until the system reasons that a conclusion cannot be reached or convergence is achieved and a conclusion regarding the visual fault is reached. The support matrix is pre-defined from training samples.

The *quality* of classification, *q*, for a fault is defined as the greatest accumulated probability over the second greatest, so that quality greater than one (q>1) constitutes a definite conclusion. Evaluation of the linear correlation algorithm on a full set of 268 artificial samples was promising, with 16 of the 19 faults achieving q>1, and 74% giving q>2, but realistic noisy images have proved much harder to analyse. To cope with the non-linearity of the system, various non-linear algorithms are proposed, and proof of principle neural networks have been tested in place of discrete parts of the linear correlation algorithm.

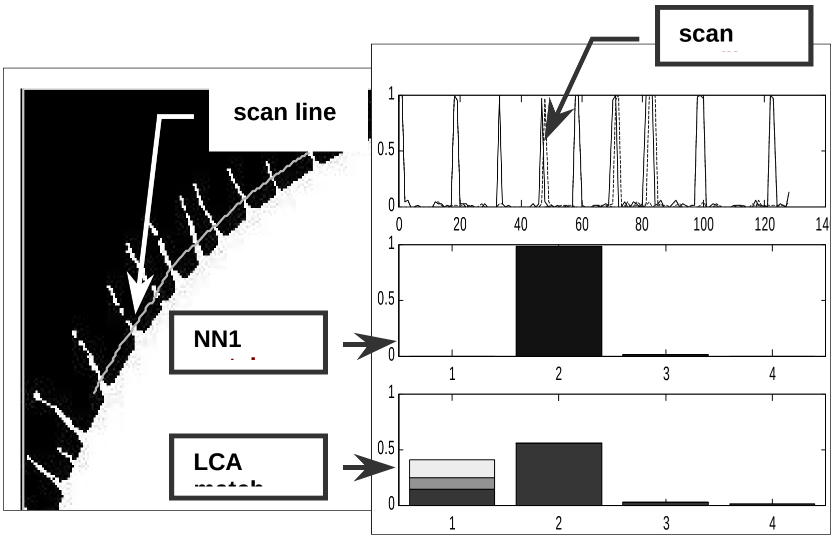

Figure 1: Comparison of the first stage of the LCA with neural network 'NN1' in classifying a scan-profile to the classes '*line*', '*hairs*', '*step*' or *'the rest'*. In this example a profile for the fault '*hairs*' is offered, to give the intermediate match *M*ij.

NEURAL NETWORKS

To facilitate comparison, the first neural network (‘NN1’) only performs the first task of the linear correlation algorithm to find the intermediate match vector *M*ij. Then the subsequent mapping using *S*ijk to probability of faults *P*ijk can be performed with the original linear correlation algorithm, but for the proof-of-principle this was not done. A feed-forward three layer net was used with a logarithmic sigmoid transfer, and trained using the back propagation with momentum. To speed up training, the match types *R*j are restricted to four models: ‘*straight line*’, ‘*step*’ and ‘*oscillatory*’ functions, and ‘*the rest*’. Initial results were encouraging with the neural network giving a higher proportional of correct fault classifications than the LCA.

Another application of a neural network (‘NN3’) is to replace the mapping from intermediate match *M*ij to *P*ijk. That is, the second stage of the LCA was compared with NN3. Again, a three layer feed-forward network is employed, which has initially been tested on 308 laboratory examples. The input match set *R* is limited to 13 so restricting the size of *M*, while the training probability-target *P*ijk is over the full catalogue, of 18 faults plus one good graphic. Thus far, the proof-of-principle network performs as effectively as the original linear correlation.

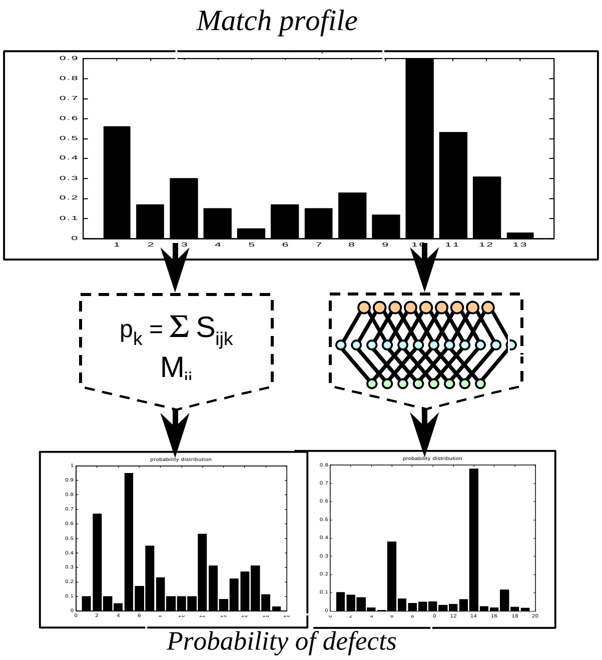

Figure 2: Comparison of the second stage of the LCA, performing *P*k = Σ *S*ijk *M*ij, with neural network ‘NN3’ in calculating the probability distribution '*P*' over the 19 faults, from the intermediate match profile '*M*' .

CONCLUSION

In order to develop an economical and flexible system for industry, a Pentium II serial processor was employed, with the MATLAB Image Processing and Neural Network Toolboxes making efficient prototyping and experimentation possible. The linear correlation algorithm gave good classification results during evaluation with 268 laboratory examples. The algorithm produces a definite conclusion for 84% of the 19 fault classes. The first neural network NN1 is promising, with a good balance of distribution to minimise errors, and easily implemented. It is reasonable for the real world scene to be mapped to an intermediate classification, with adaptive, non-linear techniques. The original linear correlation can then be employed to produce the final classification, and this may be tested in future. ‘NN3’ finds the probability of the nineteen faults from the intermediate match, *M*ij, and initial tests have proved inconclusive. Another neural network has been tested, and this classifies two-dimensional image data to the 19 faults. The advantages of neural networks include quick design and a non-linear response which will probably be required to classify faults in an operational environment. The main disadvantage is long training time although this can be reduce by modern algorithms and appropriate network architecture. These techniques will be investigated further, as more real samples from the whole gamut of visual faults are obtained. Ultimately, the success of linear and non-linear algorithms relies upon the quality of input data. Measures are being taken to improve the conditions for image acquisition and pre-processing.

Research has been undertaken with neural networks on a parallel distributed hardware architecture, as presented by Shippen, Westra and Freear (1999). In general, neural networks may be applied to classifying multi-dimensional defect spaces, for control of the printing process, and faster algorithms and processors make real-time operation possible. The results may be applied to other processes, including web printing where computer aided inspection by the operator frequently restricts production rates.

ACKNOWLEDGEMENTS

The research forms part of the Brite-Euram III Intelpadprint project BE-96-3180, funded through the European Commission. We wish to thank the EC, and the partners: Alfa Tools GmbH, the LEGO Group, Marabuwerke GmbH, Philips Electronics N.V. , and Tampoprint GmbH. MATLAB is a registered trademark of The Mathworks Inc.

REFERENCES

Beale, R. and Jackson, T., 1990, “Neural Computing: An Introduction”, London/ Bristol: Institute of Physics Publishing Ltd.

Demuth, H. and Beale, M., 1994, “Neural Network Toolbox User’s Guide: for use with MATLAB”, MA: The Mathworks Inc.

Shippen, J.M., Westra, R.L. and Freear, N.D., 1999, “A Parallel Hardware Multiple Microprocessor Asynchronous Neural Network Architecture for Control Applications”, 7th European Congress on Intelligent Techniques and Soft Computing (EUFIT), Aachen, DE