



ELSEVIER

Pattern Recognition Letters 000 (2000) 000–000

Pattern Recognition
Letters

www.elsevier.nl/locate/patrec

Application of machine learning in industrial radiographic testing

Petra Perner^{a,*}, Uwe Zscherpel^b, Carsten Jacobsen^b

^a Institute of Computer Vision and Applied Computer Sciences Leipzig, Arno-Nitzsche-Strasse 45, 04277 Leipzig, Germany

^b Bundesanstalt für Materialforschung und -prüfung, Unter den Eichen 87, 12205 Berlin, Germany

Abstract

In this paper, we are empirically comparing the performance of neural nets and decision trees based on a data set for the detection of defects in welding seams. This data set was created by image feature extraction procedures working on digitized X-ray films. We introduce a framework for distinguishing classification methods. We found that more detailed analysis of the error rate is necessary in order to judge the performance of the learning and classification method. However, the error rate cannot be the only criterion for comparing between the different learning methods. This is a more complex selection process that involves more criteria that we are describing in this paper. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Machine learning; Decision tree induction; Neural networks; Performance analysis; Radiographic testing; Pipe inspection

1. Introduction

Developing image interpretation system comprises two tasks: selecting the right features and constructing the classifier.

The selection of the right method for the classification is not easy and often depends on the preference of the system developer. This paper describes a first step towards a methodology for the selection of the appropriate classification method. Our investigation was not done on a standard academic data set where the data are usually nicely cleaned up. The basis for our investigation is an image database that contains X-

ray images from welding seams. An image is decomposed into regions of interest and, for each region of interest, 36 features are calculated by an image processing procedure. A detailed description of this process is given in Section 2. In the resulting database, each entry describes a region of interest by means of 36 feature values and a class label determined by destructive testing after the X-ray penetration. The task is to classify the Regions of Interest (ROI) automatically into background or into defects such as crack and undercut.

Two different kinds of classifier were trained based on that data set: neural nets and decision trees. The different kinds of neural nets and decision trees are described in Section 3. Since the class is given, we are dealing with supervised learning. We introduce a framework for distinguishing learning and classification methods in Section 4. A detailed description of the performance analysis is

* Corresponding author. Tel.: +49-341-8665-669; fax: +49-341-8665-636.

E-mail addresses: ibaiperner@aol.com (P. Perner),

given in Section 5. In contrast to most other work on performance analysis (e.g. Michie et al., 1994), we found that a more detailed analysis of the error rate is necessary in order to judge the performance of the learning and classification methods. However, the error rate cannot be the only criterion for the comparison between the different learning methods. It is a more complex selection process that involves more criteria such as explanation capability, the number of features involved in classification, etc. Finally, we will give our conclusions in Section 6.

2. The application

The subject of this investigation is the in-service inspection of welds in pipes of austenitic steel. Pipe systems in a power plant are inspected routinely during the lifetime of the power plant by radiographic testing to ensure the integrity of the equipment. Double wall penetration with X-rays ($E_{\max} < 200$ KeV) and special radiographic films for NDT with lead screens are used for this inspection. The flaws to be looked for in the austenitic welds are longitudinal cracks due to intergranular stress corrosion cracking starting from the inner side of the tube.

All the data were collected during a Round Robin Test (Brast et al., 1997). The last step of this Round Robin Test was a destructive testing (grinding) of the inspected welds. This gives the most advantageous situation, that the “truth” of all indications is known which is not the normal case in industrial non-destructive testing. The radiographs are digitized with a spatial resolution of $70 \mu\text{m}$ and a gray level resolution of 16 bit per pixel by a special NDT film scanner. Afterwards they are stored and decomposed into various ROI of 50×50 pixel size. The essential information in the ROIs is described by a set of features which is calculated from various image-processing methods.

These image-processing procedures are based on the assumption that the crack is roughly parallel to the direction of the weld. This assumption is reasonable because of the material and welding technique. It allows us to define a marked prefer-

ential direction. It is now feasible to search for features in gray level profiles perpendicular to the weld direction in the image. In Fig. 1, the ROIs of a crack, an undercut and of no disturbance are shown with the corresponding cross sections and profile plots.

Flaw indications in welds are imaged in a radiograph by local grey level discontinuities. Thus, it is reasonable to apply the well-known morphological edge finding operator (Klette and Zamperoni, 1992), the derivative of Gaussian operator (Pratt, 1991) and the Gaussian weighted image moment vector operator (Eua-Anant et al., 1996). These filters are developed to enhance small local gray value changes on an inhomogeneous background.

A one-dimensional FFT-filter for this special crack detection problem was designed (Zscherpel et al., 1995). This filter is based on the assumption that the preferential direction of the crack is positioned in the image in the horizontal direction. The second assumption that was determined empirically is that the half power width of a crack indication is smaller than $300 \mu\text{m}$. The filter consists of a columnwise FFT highpass Bessel operation that works with a cutoff frequency of 2 l/mm . Normally the half-power width of undercuts is greater so that this filter suppresses them. This means that it is possible to distinguish between undercuts and cracks with this FFT-filter. A row-oriented lowpass that is applied to the output of this filter helps to eliminate noise and to point out the cracks more clearly.

Furthermore, a Wavelet filter was used (Strang, 1989). The scale representation of the image after the Wavelet transform makes it possible to suppress the noise in the image with a simple threshold operation without losing significant parts of the content of the image. The noise in the image is an interference of film and scanner noise and irregularities caused by the material of the weld.

The features which describe the content of the ROI are extracted from profile plots which run through the ROI perpendicular to the weld.

In a single profile plot, the position of a local minimum is detected which is surrounded by two maxima that are as large as possible. This definition varies a little depending on the respective

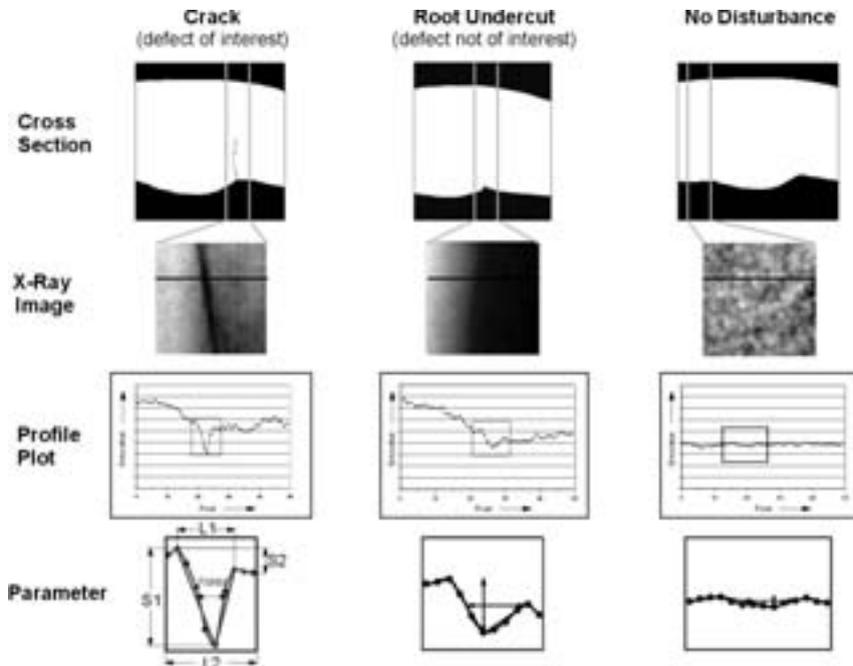


Fig. 1. ROI of crack, of an undercut and of a region without disturbance with corresponding crosssections and profile plots.

image processing routine. A template which is adapted to the current profile of the signal allows us to calculate various features ($S1, S2, L1, L2$). Moreover the half-power width and the respective gradients between the local extrema are calculated. To avoid statistical calculation errors, the calculation of the template features is averaged over all of the columns along an ROI. This procedure leads to 36 parameters.

The data set used in this experiment contains features for regions of interest from background, crack and undercut regions. The data set consists of altogether 1924 ROIs with 1024 extracted from regions of no disturbance, 465 from regions with cracks and 435 from regions with undercuts.

3. Learning methods

We have used induction of decision trees and learning of neural nets for our problem.

Four types of neural networks are used here: a Backpropagation, a Radial Basis Function Network (RBF), a Fuzzy ARTMAP Network and

Learning Vector Quantization (LVQ) Network (e.g. Zell, 1994). The learning algorithm of the network is based on the gradient descent method for error minimization. First, we tried the wrapper model for feature selection (Lui and Motoda, 1998). We did not obtain any significant result with this model. Therefore, we used a parameter significance analysis (Egmont-Petersen, 1994) for feature selection. Based on that model, we can reduce the parameters to seven significant ones.

For decision tree induction we used our software package called Decision Master. An entropy minimization criterion is used for attribute selection (Quinlain, 1996) and reduced-error pruning technique (Quinlain, 1987a) is used for tree pruning. Numerical attribute discretization is done based on methods described in Perner and Trautzsch (1998). Decision tree induction may also be looked upon as a method for attribute selection. During the learning phase, only the most relevant attributes are chosen from the whole set of attributes for the construction of decision rules in the nodes. Therefore, we do not need to carry out

feature selection before the learning process as in the case of neural nets.

4. Evaluation criterion

The evaluation criterion most used for a classifier is the error rate $f_r = N_f/N$ with N_f the number of falsely classified samples and N the whole number of samples. In addition to that, we use a contingency table in order to show the qualities of a classifier, see Table 1. In the field of the table are input the real class distribution and the class distribution proposed by the classifier as well as the marginal distribution c_{ij} . The main diagonal is the number of correctly classified samples. The last row shows the number of samples assigned to the class shown in row 1 and the last line shows the class distribution proposed by the classifier.

From this table, we can calculate parameters that assess the quality of the classifier.

The *correctness*

$$p = \frac{\sum_{i=1}^m c_{ii}}{\sum_{i=1}^m \sum_{j=1}^m c_{ij}}$$

is the number of correctly classified samples according to the number of samples. That measure is the opposite of the error rate.

For the investigation of the classification quality we measure the *classification quality* $p_{ii} = c_{ii} / \sum_{i=1}^m c_{ij}$ according to a particular class i and the *class specific quality* that is $p_{ki} = c_{ii} / \sum_{j=1}^m c_{ji}$ the number of correctly classified samples for one class. In addition to that we use other criteria shown in Table 2.

One of these criteria is the cost for classification expressed by the number of features and the number of decisions used during classification. The other criterion is the time needed for learning. We also take the explanation capability of the classifier into consideration as another quality criterion. It is also important to know if the classification method can learn the classification function (the mapping of the attributes to the classes) correctly based on the training data set. Therefore, we not only consider the error rate based on the test set we also consider the error rate based on the training data set. For the evaluation of the error rate, we used the test and train method instead of cross-validation (Weiss and Kulikowski, 1991) since it would have been computationally expensive to evaluate the neural nets by crossvalidation.

Table 1
Contingency table

Assigned class index	Real class index				
	1	i	...	m	Sum
1	c_{11}	c_{1m}	
j	...	c_{ji}	
...	
m	c_{m1}	c_{mm}	
Sum					

5. Results

5.1. Error rate, generalization and representation ability

The error rate for the design data set and the test data set is shown in Table 3. The unpruned decision tree shows the best error rate for the design data set. This tree represents the data best.

Table 2
Criteria for comparison of learned classifiers

Generalization capability of the classifier	Error rate based on the test data set
Representation of the classifier	Error rate based on the design data set
Classification costs	Number of features used for classification
	Number of nodes or neurons
Explanation capability	Can a human understand the decision?
Learning performance	Learning time
	Sensitivity to class distribution in the sample set

Table 3
Error rate for design data set and test data set

Name of classifier	Error rate on design data set in %	Error rate on the test data set in %
Binary decision tree		
Unpruned	1.7	9.8
Pruned	4.6	9.5
<i>n</i> -ary decision tree		
Unpruned	8.6	12.3
Pruned	8.6	12.3
Backpropagation network	3.0	6.0
RBF neural net	5.0	5.0
Fuzzy-ARTMAP-Net	0.0	9.0
LVQ	1.0	9.0

However, if we look for the error rate calculated based on the test data set, then we note that the RBF neural net and the Backpropagation network can do better. Their performance shows not such a big difference in the two error rates as that for the decision tree. The representation and generalization ability is more balanced in the case of the neural networks whereas the unpruned decision tree gets overfits the data. This is a typical characteristic of decision tree induction. Pruning techniques should reduce this behavior. The representation and generalization ability of the pruned tree shows a more balanced behavior but it cannot outperform the results of the neural nets.

The behavior of the neural nets according to their representation and generalization ability is controlled during the learning process. The training phase was regularly interrupted and the error rate was determined based on the training set. When the error rate decreased after a maximum value, then the net was assumed to have reached its maximal generalization ability.

It is interesting to note that the Fuzzy-ARTMAP-Net and the LVQ have the same behavior with respect to their representation and generalization ability as the decision trees.

The observation for the decision tree suggests another methodology for using decision trees. In data mining, where we mine a large database for the underlying knowledge it might be more appropriate to use decision tree induction since it can represent the data well.

The performance of the RBF expressed by the error rate is 4% better than it is for the decision tree. The question is: How can we access this result? Is 4% a significant difference? We believe the decision must be made based on the application. In some cases it might be necessary to have a 4% better error rate whereas in other cases it might not have a significant influence.

5.2. Classification quality and class specific quality

We obtain a clearer picture of the performance of the various methods if we look for the classification quality p_k and the class-specific classification quality p_i , see Tables 4–8. In the case of the decision tree, we observe that the class specific quality for class undercut is very good and can

Table 4
Contingency table of the classification result for the decision tree

Classification result	Real class index			
	Background	Crack	Undercut	Sum
Background	196	2	1	199
Crack	12	99	22	132
Undercut	0	1	58	59
Sum	208	102	80	390
p_k	0.94	0.97	0.73	
p_i	0.98	0.74	0.98	

$p = 0.90$
 $k = 0.85$

Table 5
Contingency table of the classification result for the back-propagation net

Classification result	Real class index			
	Background	Crack	Undercut	Sum
Background	194	5	1	200
Crack	1	97	3	100
Undercut	13	0	76	90
Sum	208	102	80	390
p_k	0.93	0.95	0.95	
p_i	0.97	0.96	0.85	

$p = 0.94$
 $k = 0.90$

Table 6
Contingency table of the classification result for the radial basis function net

Classification result	Real class index			Sum
	Background	Crack	Undercut	
Background	198	2	1	201
Crack	1	100	7	108
Undercut	9	0	72	81
Sum	208	102	80	390
p_k	0.95	0.98	0.90	
p_l	0.99	0.93	0.89	
				$p = 0.95$ $k = 0.92$

Table 7
Contingency table of the classification result for the LVQ

Classification result	Real class index			Sum
	Background	Crack	Undercut	
Background	183	2	0	185
Crack	7	100	7	114
Undercut	18	0	73	91
Sum	208	102	80	390
p_k	0.88	0.98	0.91	
p_l	0.99	0.87	0.80	
				$p = 0.91$ $k = 0.86$

Table 8
Contingency table of the classification result for the fuzzy-ARTMAP net

Classification result	Real class index			Sum
	Background	Crack	Undercut	
Background	191	5	1	197
Crack	10	96	10	116
Undercut	7	1	69	77
Sum	208	102	80	390
p_k	0.92	0.94	0.86	
p_l	0.97	0.83	0.90	
				$p = 0.91$ $k = 0.86$

outperform the error rate obtained by neural nets. In contrast to that, the classification quality for decision trees is poorer. Samples from the class

“undercut” and “background” are falsely classified into the “crack” class. That results in a low classification quality of the class crack. In contrast to that, the neural nets show a difference in class specific recognition. Mostly, the class specific error for class crack is considerably better than for the decision trees (e.g. 0.96 for backpropagation net in Table 5 compared with 0.74 for the decision tree in Table 4). Since a defect crack is more important than the class undercut (which is not a defect but a geometrical indication from the production process of the pipe) it would be better to have the lowest class specific error for crack. This avoids false alarms and can save a lot of repair costs in real life.

5.3. Classification cost

Our decision tree produces decision surfaces that are parallel to the feature axes. This method should be able to approximate a non-linear decision surface, of course with a certain approximation error, which will lead to a higher error rate in classification. Therefore, we are not surprised that there are more features involved in the classification, see Table 9. The unpruned tree uses 14 features and the pruned tree uses 10 features. The learned decision tree is a very big and bushy tree with 450 nodes. The pruned tree reduces to 250 nodes but still remains a very big and bushy tree.

The neural nets work only on seven features and e.g. for the backpropagation net we have 72 neurons. That reduces the effort for feature extraction and computation of the classification result drastically.

Table 9
Number of features and number of nodes

Name of classifier	Number of features	Number of nodes
Decision tree		
Unpruned	14	450
Pruned	10	250
Backpropagation net	7	72 neurons
RBF net	7	
LVQ	7	51 neurons
Fuzzy-ARTMAP-Net	7	

5.4. Learning performance

One of the advantages of induction of decision trees is the fact that they are easy to use. Feature selection and tree construction is done automatically without human interaction. It takes only a few seconds on a PC 486 until the decision tree has been learned for the data set of 1924 samples used. In contrast to that, neural network learning cannot be handled as easily. The learning time for the neural nets was 15 min for the backpropagation net by 60 000 learning steps, 18 min for the RBF net, 20 min for the LVQ net, and 45 min for the Fuzzy-ARTMAP-net on a workstation type INDY Silicon Graphics (R4400-Processor, 150 MHz) with the neural network simulator Neural-Works Professional II/Plus version 5.0.

A disadvantage of neural nets is that the feature selection is a preliminary step before learning. In our case it was done with a backpropagation net. The parameter significance was determined based on the contribution analysis method (see Section 3). From 42 features are selected only the seven most significant features which were selected by the feature selection process. If the feature selection process is omitted, which is possible, then the training time and the size of the neural nets increase considerably.

Although the class distribution in the design data set was not uniform, both methods of neural net-based learning and decision tree learning did well on the data set. It is possible that in the case of decision trees, the poorer class specific error rate for crack results from the non-uniform sample distribution. Therefore, we used a data set in one experiment where the number of samples for each of the three classes was the same (420 samples). As a result we got an error rate that was nearly the same as the one shown in Table 3.

5.5. Explanation capability

One big advantage of decision trees is their explanation capability. A human can understand the rules and can control the decision making process. A neural net based classifier is more or less a black box for humans. However, since the outcome of the learning process is a very bushy tree, this rep-

resentation is not very readable for humans. Therefore, some researchers favor rule-based representations over tree structured representations since rules tend to be more modular and can be read in isolation of the rest of the knowledge base constructed by induction. A compromise is to use decision tree induction to build an initial tree and then derive rules from the tree, thus transforming an efficient but opaque representation into a transparent one (Quinlain, 1987b).

However, investigation in (Perner et al., 1996) showed that the model derived by decision tree induction does not always represent the model of a human expert. The decisions might not appear in the order a human would use them and especially not at all in uncertain domains.

6. Conclusions

We compared decision tree induction to neural nets. For our empirical evaluation, we used a data set of X-ray images that were taken from welding seams. Thirty six features described the defects contained in the welding seams. Image feature extraction was used to create the final data set for our experiment.

We found that special types of neural nets have slightly better performance than decision trees. In addition to that, there are not so many features involved in the classification, and this is a good characteristic especially for image interpretation tasks. Image processing and feature extraction is mostly time-consuming and requires special purpose hardware for real time processing for the computation. However, the explanation capability that exists for trees producing axis-parallel decision surfaces is an important advantage over neural nets. Moreover, the learning process is not so time-consuming, it comprises the two processes necessary when building image interpretation systems: feature selection and learning the classifier. Furthermore, it is easy to handle.

All that shows that the decision on what kind of method should be used is not only a question of the resulting error rate. It is a more complex selection process that can only be based on the constraints of the application. The present paper

provides some new aspects to the selection of a classification algorithm for a particular application.

References

- Brast, G., Maier, H.J., Knoch, P., Mletzko, U., 1997. Progress Report on a NDT Round Robin on Austenitic Circumferential Pipe Welds. Proceedings 23, MPA Seminar, Vol. 2. Stuttgart, Germany, pp. 42.1–42.16.
- Egmont-Petersen, L., 1994. Contribution Analysis of multi-layer perceptrons. Estimation of the input sources importance for the classification. In: Pattern Recognition in Practice IV, Proceedings International Workshop. Vlieland, The Netherlands, pp. 347–357.
- Eua-Anant, N., Elshafey, I., Upda, L., Gray, J. N., 1996. A novel image processing algorithm for enhancing the probability of detection of flaws in X-ray images. In: Review of Progress in Quantitative Non-destructive Evaluation, Vol. 15, Plenum Press, New York, pp. 903–910.
- Klette, R., Zamperoni, P., 1992. Handbuch der Operatoren für die Bildbearbeitung. Vieweg Verlagsgesellschaft.
- Lui, H., Motoda, H., 1998. Feature Selection for Knowledge Discovery and Data Mining. Kluwer Academic Publishers, Dordrecht.
- Michie, D., Spiegelhalter, D.J., Taylor, C.C., 1994. Machine Learning, Neural Nets and Statistical Classification. Ellis Horwood Series in Artificial Intelligence, Chichester, UK.
- Perner, P., Belikova, T.B., Yashunskaya, N.I., 1996. Knowledge acquisition by decision tree induction for interpretation of digital images in radiology. In: Perner, P., Wang, P., Rosenfeld, A., (Eds.), Advances in Structural and Syntactical Pattern Recognition, Lecture Notes on Computer Science, Vol. 1121. Springer, Berlin, pp. 208–219.
- Perner, P., Trautzsch, S., 1998. On feature partitioning for decision tree Induction. In: Amin, A., Dori, D., Pudil, P., Freeman, H., (Eds.), SSPR98 and SPR98. Springer, Berlin, pp. 475–482.
- Pratt, K.W., 1991. Digital Image Processing. Wiley, New York, ISBN 0-471-85766-1.
- Quinlain, J.R., 1996. Induction of decision trees. Machine Learning 1, 81–106.
- Quinlain, J.R., 1987a. Simplifying decision trees. Int. J. Man-Machine Studies 27, 221–234.
- Quinlain, J.R., 1987b. Generating production rules form decision trees. In: Proceedings of the 10th International Joint Conference on Artificial Intelligence. Morgan Kaufmann, San Mateo, CA, pp. 304–307.
- Strang, G., 1989. Wavelets and Dilation Equations: A Brief Introduction Sam Review, Vol. 31, pp. 613–627.
- Weiss, S.M., Kulikowski, C.A., 1991. Computer Systems that Learn. Morgan Kaufmann, Los Altos, CA.
- Zell, A., 1994. Simulation Neuronaler Netze. Addison-Wesley, Bonn Paris.
- Zscherpel, U., Nockemann C., Mattis A., Heinrich W., 1995. Neue Entwicklungen bei der Filmdigitalisierung. DGZfP-Jahrestagung in Aachen, Tagungsband.