ELSEVIER

PII: S0963-8695(96)00045-X

NDT&E International, Vol. 30, No. 1, pp. 3-10, 1997

0963-8695/97 $17.00+0.00

Optimization of the recognition of defects in flat steel products with the cost matrices theory

Florent Dupont*, Christophe Odet*, Michel Carton†

*CREATIS, CNRS Research Unit (UMR 5515), affiliated to INSERM, INSA 502, 69621 Villeurbanne Cedex, France
†IRSID, Voie Romaine, BP 320, 57214 Maizières les Metz, France

Received 20 May 1996; revised 31 July 1996

There is an urgent need for automatic systems for quality control of flat steel products to meet the ever increasing demand from customers, and for process control to prevent damage to production. To achieve these goals, those inspection systems should detect and recognize defects using image processing and pattern recognition methods. Such complex systems are difficult to compare and the specificity of each application is generally not taken into account. This paper describes a method for optimizing a system using a cost matrix approach. It produces an image which can be used to appreciate the quality of the classifier visually. We have applied the method in order to optimize the feature selection and to compare the classification methods used in a system designed for recognizing defects in flat steel products in a cold rolling mill. The results show significant improvements in the performance of the recognition system.

Keywords: surface inspection system, pattern recognition, classifier evaluation, cost matrix

There is an urgent need for automatic systems for quality control of flat steel products to meet the ever increasing demand from customers, the high cost of correction of poor quality and the need to reduce the stress on inspectors. This means that systems are required to provide on-line defect detection and recognition or identification. The results provided by such systems can be used, for example, to help in the characterization of the industrial process and to carry out corrections to plant equipment and hence ensure customer satisfaction.

Information on defect identity and cause is needed in near real-time for remedial action, but quality appraisal of processed material can be relaxed to just in time for decision making on the suitability for the customer or further processing.

Throughout the 1980s, machine vision systems for industry were developed first by the automobile manufacturers and then for all the branches of manufacturing industry. Product quality receives more and more attention, and image processing hardware has made significant progress. These reasons led companies to take an interest in the automatic inspection problem. In 1984, nine iron and steel making companies and three aluminium companies carried out a research project called the American Iron and Steel Institute's Surface Inspection Project. The project was realized by the Westinghouse Electric Corporation[1]. After collaborating with Eastman Kodak, a prototype of an inspection system was built and tested at different steel and aluminium production lines in 1987. They showed that the industrialization of an inspection system was possible. At the same time, another company, Litton Integrated Automation[2], built a system with CCD sensors to identify defects. From the team which worked on this project, two companies were born: Isys Control Inc. and Qualimatrix. They offer different systems including image processing and classification. At the same time, European companies began to work on this problem, including European Electronic Systems Limited, Fabricom, Rautaruukki O.Y.[3] and others[4,5].

For all manufacturers, the installation of an inspection system is expensive. The decision to buy such a system needs insurance that there will be a better quality product and/or cost reduction in the process line. Indeed, some factories prefer to rent a system to test the accuracy of the inspection system and estimate the benefit it will bring. Some difficulties often appear:

- the specificity of each factory is not sufficiently taken into account;
- the accuracy is very difficult to evaluate, and a comparison between two systems is not easy. So the need to have a method in order to evaluate the real possibilities of an inspection system is very urgent. Moreover, these systems should be more adapted to the application in a more automatic way.

In regard to these remarks, we think it is necessary to define a criterion, adapted to one application, in order to optimize the accuracy of the inspection system and to have a measure to be able to compare and evaluate these systems. The aim of this paper is to describe such a criterion.

In the following, we describe the industrial application for which we have developed an inspection system. Then, the cost matrices theory is proposed with a review of previous papers written on this problem. Finally, we present the results we have obtained for the application and the benefit of using our method.

Industrial application

The method we propose in this article has been applied in the iron and steel industry for the recognition of defects in flat steel products in a cold rolling mill[6]. We have implemented the scheme at a plant located in Florange near Metz in France, where the pickling line is linked to the cold rolling line by an accumulator in order to have enough time to prevent damage during rolling.

At the present time, the system is capable of detecting surface flaws. The operator has a video monitor on which the image of the moving steel strip is scrolled. When a defect is detected by the system, the suspect area is frozen on the screen. The action of the operator depends on the type of defect and its severity: he can slow the line down or open the rolling mill stands. At high production speed (10 m/s) human inspection becomes very difficult. The prototype detection system developed by IRSID (French Institute of Research in Iron and Steel)[7] consists of CCD linear cameras, halogen light sources and a dedicated electronic system. Two cameras inspect both sides of the sheet metal simultaneously and another camera detects the edges of the strip and possible holes. Images are reconstructed and processed in real-time in order to detect defects.

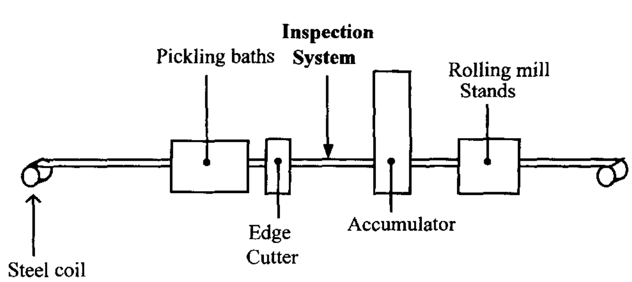


Figure 1 Location of the inspection system in the pickling-cold rolling plant

The defect detection system provides good results because all the defects produce an alarm. However, the level of false alarms (about 80%) is too high (an alarm is produced by the system even if a defect is not present on the steel strip). In order to reduce the workload of the operator the installation of a defect recognition system is highly desirable.

Defect identification serves two purposes:

- production facility protection: defects that can damage the rolls or cause production stoppages must be detected and identified;
- production quality must comply with customer requirements; the system provides both process management and quality control with regard to the customer. When a non-conformance is detected, the operator can take steps to correct the manufacturing parameters causing the defect.

Over several weeks, a defect recording system was installed and more than 3000 images were stored on magnetic tapes in order to build a large database of images for classification purposes.

Cost matrices theory

A recognition system is composed of many processing stages. A global optimization procedure is needed in order to reach the goals that have been fixed for the application. As part of this effort, we are interested in methods that can be used to measure the accuracy of the system. The system takes a decision and associates a class to each defect. Unfortunately a simple recognition is not sufficient to take into account the specificities associated with most real problems. Indeed, the severity of a misclassification error should not be considered the same for each class of defects. This can be accomplished using confusion matrices.

Confusion matrices

Classification results are stored in a matrix $(Conf)$ of size $K \times K$ where K is the number of classes. Each element $Conf_{i,j}$ represents the probability of a defect belonging to class i being recognized as an element of class j. For each class, we have:

$$\forall i \in [1, K] \sum_{j=1}^{j=K} Conf_{i,j} = 1 \tag{1}$$

Diagonal terms of a confusion matrix correspond to elements which have been well classified by the system.

So a perfect system should produce a confusion matrix with only diagonal terms. A high value $Conf_{i,j}$ means that classes i and j are not separated in feature space. But, if the number of classes is high, it becomes very difficult to analyse the confusion matrix.

A considerable amount of work has been done relating to issues of estimating classifier performance[8]. Many of these utilize methods to find the best discrimination rule. Vallet[9] normalizes confusion matrices in order to have the same diagonal elements. Comparison between different classifiers therefore becomes easier. However, these methods are based on confusion matrices, and consequently do not allow us to compare or to evaluate many recognition systems when misclassification rates are not of the same order for each class.

Introduction of a measure to evaluate and optimize recognition systems

In applications where each class has a different severity, we will show how a cost matrix can be used following a confusion matrix, in order to evaluate a recognition system.

The first stage consists of establishing a cost matrix ($Cost$) of size $K \times K$ where K is the number of classes, written for the application. Each element $Cost_{i,j}$ represents the cost to misclassify an element of class i as class j.

Then, we compute the total cost of misclassifications of a recognition system with the following formula:

$$Cost_{total} = \sum_{i=1}^{i=K} \sum_{j=1}^{j=K} N_{i,j} \times Cost_{i,j} \tag{2}$$

where

$$N_{i,j} = N_i \times Conf_{i,j} \tag{3}$$

and N_i is the size of class i.

The best system for one application (or one cost matrix) has the minimal value of $Cost_{total}$. So the only value of this measure allows one to compare different classification approaches. If you introduce a loop in the processing scheme of the recognition system, you are able to optimize it. $Cost_{total}$ is the mathematical criterion used to optimize the recognition, by giving priority to classes of defects that are important for the application, i.e. defects with high severity. We will illustrate this statement in the next part.

How to write the cost matrix

As we have already said, for a large number of applications, the introduction of a cost matrix is necessary to evaluate a classification system. To write this cost matrix, we propose an approach with an expert of the application domain. This stage becomes a very important one in the construction of the system.

First of all, classes have to be ordered in a hierarchy. For our application, the intrinsic severity of defects has been used. Let us see how a numeric cost for each confusion type can be determined.

First stage

Let K be the number of classes. Classes with the same severity are grouped into a family. A family can contain one or more classes. Let f be the number of families.

Second stage

An integer matrix called $Order$ of size $f \times f$ is built. The row index i represents the true family of defects, and the column index j represents the family identified by the recognition system. If a family i contains only one class of defects, the element $Order_{i,i}$ is empty (see element 1, 1 with the minus sign in Table 2) because there is no possible confusion within family i. Then, for each other couple i, j, the severity of the misclassification of a defect belonging to family i recognized as a defect in family j (j could be equal to i) is used to determine the integer value $Order_{i,j}$. The $Order_{i,j}$ value reflects the severity of the misclassification. So, the first $Order$ value 1 is given for the more severe misclassifications. For example, in Table 2, $Order_{1,3} = 1$ means that the recognition of a family 1 defect as a family 3 defect has a high severity. The following $Order$ value 2 is given to a less severe misclassification, and so on. The procedure is repeated until all the elements are filled. The same value can be given to several elements $Order_{i,j}$. The last integer order value M is given to the lowest severity confusion. For example, in Table 2, $Order_{3,3} = 8$ means that the recognition of a family 3 defect as a different family 3 defect has a very low severity. The values $Order_{i,j}$ are integers with $1 \leq Order_{i,j} \leq M$ with $M \leq f^2$. The case where $M = f^2$ arises when all the elements $Order_{i,j}$ are filled and different.

Third stage

This stage consists of defining groups p in the order matrix, with $1 < p < P$. This allows us to give a very strong hierarchy between misclassifications. P is the total number of groups and only depends on the expert wishes. If the expert does not want to create this hierarchy, P is equal to one. The misclassification of one element in group p should be more critical than the misclassification of all the elements in the groups $p+1, p+2, \ldots, P$. In this way, this rule allows us to avoid some confusions. As an example, in Table 3, the group 1 indicates that we should avoid a recognition of a family 1 defect as a family 2 or 3 defect and the recognition of a family 2 defect as a family 3 defect. This certainly corresponds to very high severity defects for the application.

Fourth stage

The final cost matrix will correspond to confusion costs between each family, so the size will be $f \times f$. For each value of $Order_{i,j}$, we compute a different cost. We apply an arbitrary cost I (for example $I = 1$) for the most important misclassification which is the first in the *Order* list ($Order_{i,j} = 1$). The following $Cost_{i,j}$ values within the same group should respect the *Order* matrix. The application expert must choose the ratio α between two consecutive costs in the same group p. α must respect the following inequality:

$$0 < \alpha \leq 1$$

A value 1 means that all the misclassifications within the group are equivalent. The cost for $Order_{i,j} = m$ is defined by:

$$Cost_{i,j}(m) = \frac{I\alpha^{(m-p)}}{N_t^{(p-1)}} \tag{4}$$

where N_t is the size of the test set used for the optimization procedure and p is the group number.

We have now obtained all the coefficients in the cost matrix. This matrix can be used to compare results from different classification methods.

Use of the cost matrix for feature selection and classifiers comparison

In an industrial context, there are two kinds of control, process control and quality control:

- The aim of process control is to identify mainly the defects that can damage the production line. So, the cost matrix should be established with a strong priority for this kind of defect. The families, which we have defined, will contain classes with the same risk of causing damage to the process.
- An objective of quality control will lead us to construct a system able to guarantee to each customer an optimal recognition rate for particular defects that interest this customer in priority. All kinds of defect can be identified with a different cost, so a specific cost matrix can be written for each customer. The use of this method, combined with a defect cartography, allows us to optimize the assignment of each product to customers.

Whatever goal we have to reach, a single criterion reveals the accuracy of a recognition system. The comparison of several systems on a test set becomes easier. In a first approach, the minimal value of the global cost computed on each confusion matrix shows the best recognition system.

The use of this criterion allows us to compare each stage of the processing scheme. For example, different classification methods can be applied on the same test set, with the same features. Then, the results are compared with the criterion.

We have applied this method in order to optimize the feature selection stage with a classification method. This stage is a very important one in a recognition system. To obtain the best feature combination, we should minimize the global misclassification cost. To find the optimal feature combination, you have to compare $2^p - 1$ combinations where p is the number of features. If you want to reduce the number p to a number n of features, you have to compare $C_n^p = p!/[n!(p-n)!]$ combinations. Some algorithms, for example the Branch and Bound algorithm[10], are used to find the optimal solution with fewer trials, by using a feature hierarchy like a tree. Nevertheless, if the number of features is high, the time to find the optimal solution is prohibitive.

An iterative selection method[11] allows us to find a suboptimal solution. Let us briefly describe the principle:

Starting point

- initial selection of features,
- classification of the data test set,
- computation of the total cost.

One iteration

If there are features that have not been used:

- addition of one feature among the others,
- classification of the data test set,
- computation of the total cost.

The feature that gives the minimal cost is added to the list of selected features.

The algorithm iterates until all features are used.

This method reduces the number of trials. If p is the total number of features, d the number of features initially selected, N is the total number of trials:

$$N = \frac{(p-d)}{2} \times (p-d+1) \tag{5}$$

The minimal cost value gives the best feature combination for the criterion based on the cost matrix. To analyse all the N results more precisely, we have proposed a method, based on a visualization of an image with all costs. We will show later some advantages of this analysis. First of all, we place all results in a triangular table, where one side represents the number of features used and on the other side, the cost values are ordered. At each stage, for a selection of n features, a column contains $p - n + 1$ cost values of classifiers that are ordered.

Example

$p = 8$ total number of features
$d = 3$ initial number of features

Then, we perform a quantization of all cost values and attribute proportional grey levels to costs. The cost value

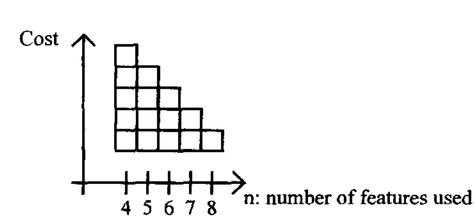


Figure 2 Ordered cost values

is represented like a third dimension and the visualization of the perspective of the cost image allows many interesting observations:

- the minimal value corresponds to the best feature selection,
- the profile of the first horizontal line corresponds to the best results at each stage of the algorithm, with an increasing number of features used from d to p,
- the study of the topology of this image can reveal more information such as the more robust areas with uniform cost values or noisy areas with irregular cost values.

A more precise analysis of the cost results allows us to choose a different classification method from the one obtained by the automatic global criterion. For example, if the application needs a reduced number of features, due to real-time constraints, the cost image facilitates the choice of the selection for the classification method used.

Application example

For our application of identification of surface defects on pickled steel sheet, we have applied this method to the database of defect images that we have constructed. The image processing transforms images from the acquisition system into defect images. Then, features are computed on these segmented defects. We now describe the different stages of image processing.

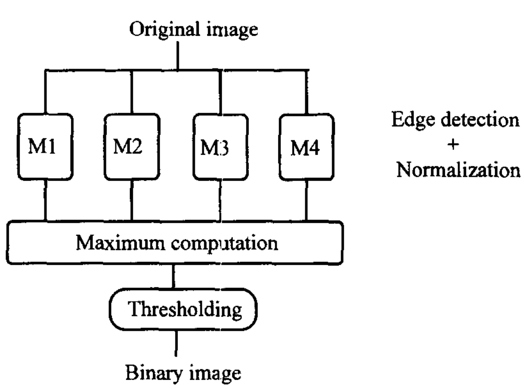


Figure 3 Principle of edge detection

Image processing

First of all, a geometrical correction is performed on images in order to redress the mean profile because the quantity of light caught by the camera depends on the angle of incidence. Then, an edge detection is applied on images. The operator chosen is the Prewitt filter, used in four directions with the following masks:

$$\text{Masks:}\quad \underset{M1}{\begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix}} \underset{M2}{\begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}}$$

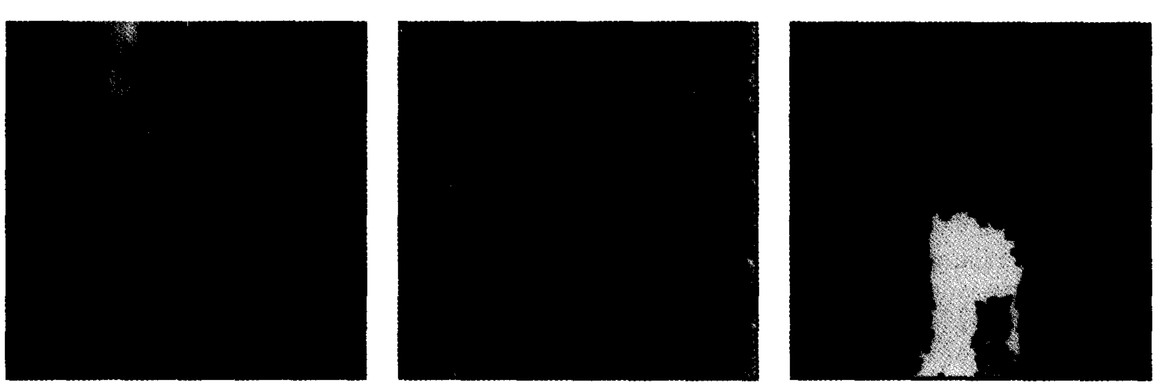
$$\underset{M3}{\begin{pmatrix} 2 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -2 \end{pmatrix}} \underset{M4}{\begin{pmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{pmatrix}}$$

Results are normalized and the maximum is determined at each point, and is then thresholded. Only the stronger edges are detected and stored in the image.

Then, morphological filters clean the noise in the edge image. The first stage is an elimination of isolated points. After, we apply a morphological closing in order to connect the edges which are close enough. The edges are closed by this operation, which is a combination of a dilation and an erosion. We have used a square kernel. The edges are filled to obtain the regions of defects instead of edge lines.

The image of defects is labelled with an algorithm which looks for connected components. Each defect corresponds to one object that is characterized by a vector of 51 features. Different kinds of features are computed: geometrical features and features based on grey level distribution.

During several weeks, the prototype of the defect



Original image | Edge image | Segmented image

Figure 4 Images during the segmentation process detection system automatically stored about 3000 images that have produced an alarm. We have selected 400 representative images with a uniform background which is the more usual texture, about 80% of the production.

First of all, an expert in the domain identified each defect of the database and a defect list was established, with all the characteristics of each class of defects. Then, all the defect images were automatically segmented with the method that we described before. We labelled all the segmented defects and obtained a defect database. The sizes of the database for each class of defects are grouped in a table. This database is split into a learning set and a test set, in order to construct the classifier and evaluate its accuracies. 40% of defects are placed in the learning set and 60% in the test set.

Each class number corresponds to a type of defect.

Table 1 Size of the defect database for each class

Bases / Classes	Learning	Test	Total
Skin lamination	222	333	555
Coil break	175	263	438
Spot	43	61	108
Shoe mark	66	101	167
Fleck scale	6	11	17
Fold	7	11	18
Pinch mark	3	5	8
Rinsing stain	85	129	214
Total	607	918	1525

Cost matrix construction

For our application, the aim of the recognition system is process control. The cost matrix should take into account the severity of defects in priority.

First stage
An expert in the domain had grouped the eight defect classes ($K = 8$) in three families ($f = 3$) for three different severities of misclassification:

- Family 1: defects that will certainly produce damage:

  class 'Pinch mark'

- Family 2: defects that could produce damage:

  class 'Skin lamination' and
  class 'Fold'

- Family 3: defects that are not dangerous for the production line:

  class 'Coil break',
  class 'Spot',
  class 'Shoe mark',
  class 'Fleck scale' and
  class 'Rinsing stain'

Second stage
The expert defined an ordered list between the different families. In this case we obtain $M = 8$. This list is given in the order matrix in Table 2, where the row index represents the true family and the column index represents the recognized family.

Table 2 Order matrix for inter-family costs

Families	1	2	3
1	–	3	1
2	4	7	2
3	5	6	8

Family 1 contains only one class, so there is no possible confusion within this family ($Order_{1,1}$ is empty).

Third stage
Then, the expert defined three groups ($P = 3$) in the order matrix in order to satisfy our initial objective (see Table 3):

Table 3 Groups in the order matrix

Families	1	2	3
1	-	3	1
2		7	2
3			8

- Group 1: $Order_{i,j} = 1, 2, 3$
- Group 2: $Order_{i,j} = 4, 5, 6$
- Group 3: $Order_{i,j} = 7, 8$.

Fourth stage
The size of the test set is $N_t = 1525$. We chose the initial cost $I = 1$ and $\alpha = 1/2$ the ratio between two successive costs in a same group. The values of the cost matrix can be computed for each value of $Order_{i,j}$, for each family confusion as explained previously (Equation (4)).

Table 4 Numerical cost matrix by family

Family	1	2	3
1	0	0.25	1
2	1.56×10^{-4}	2.44×10^{-8}	0.5
3	7.81×10^{-5}	3.91×10^{-5}	1.22×10^{-8}

Classification results

The method proposed was tested on the database of our application with the following classification methods:

- Reilly, Cooper and Elbaum method[12] (RCE),
- K nearest neighbours method[13,14] (KNN).

The RCE method is based on multilayer neural networks with an incremental architecture. A very simple initial network is progressively completed with new neurons during the learning stage. An iterative algorithm is applied until the stability of the network is obtained.

The KNN method does not need a learning stage. The class of an unknown defect is directly obtained from the computation of the distance between this defect and each known defect in the database. Among the K nearest neighbours, the majority class is affected to the unknown defect.

Those supervised methods need for input the feature vectors and the true class of each defect in the learning database.

We have defined the following values:

- the minimal cost that corresponds to a perfect classification system:

$$Cost_{min} = 0 \tag{6}$$

- the maximal cost that corresponds to the worst classification system:

$$Cost_{max} = \sum_{i=1}^{i=f} N_i \times \underset{j\in[1,f]}{\mathrm{Max}} [Cost_{ij}] \tag{7}$$

where i represents the family, f is the total number of families, and N_i is the size of the test set for family i.

We obtain $Cost_{max} = 177.04$.

- the cost value for a random classification:

$$Cost_{rand} = \sum_{i=1}^{i=f} \left(\frac{N_i}{f} \times \sum_{j=1}^{j=f} Cost_{ij} \right) \tag{8}$$

We obtain $Cost_{rand} = 110.94$.

Those values can be used as a reference in order to evaluate the classifier.

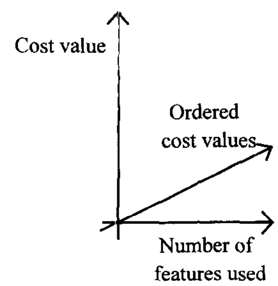


Figure 5 Principle of the 3D cost representation

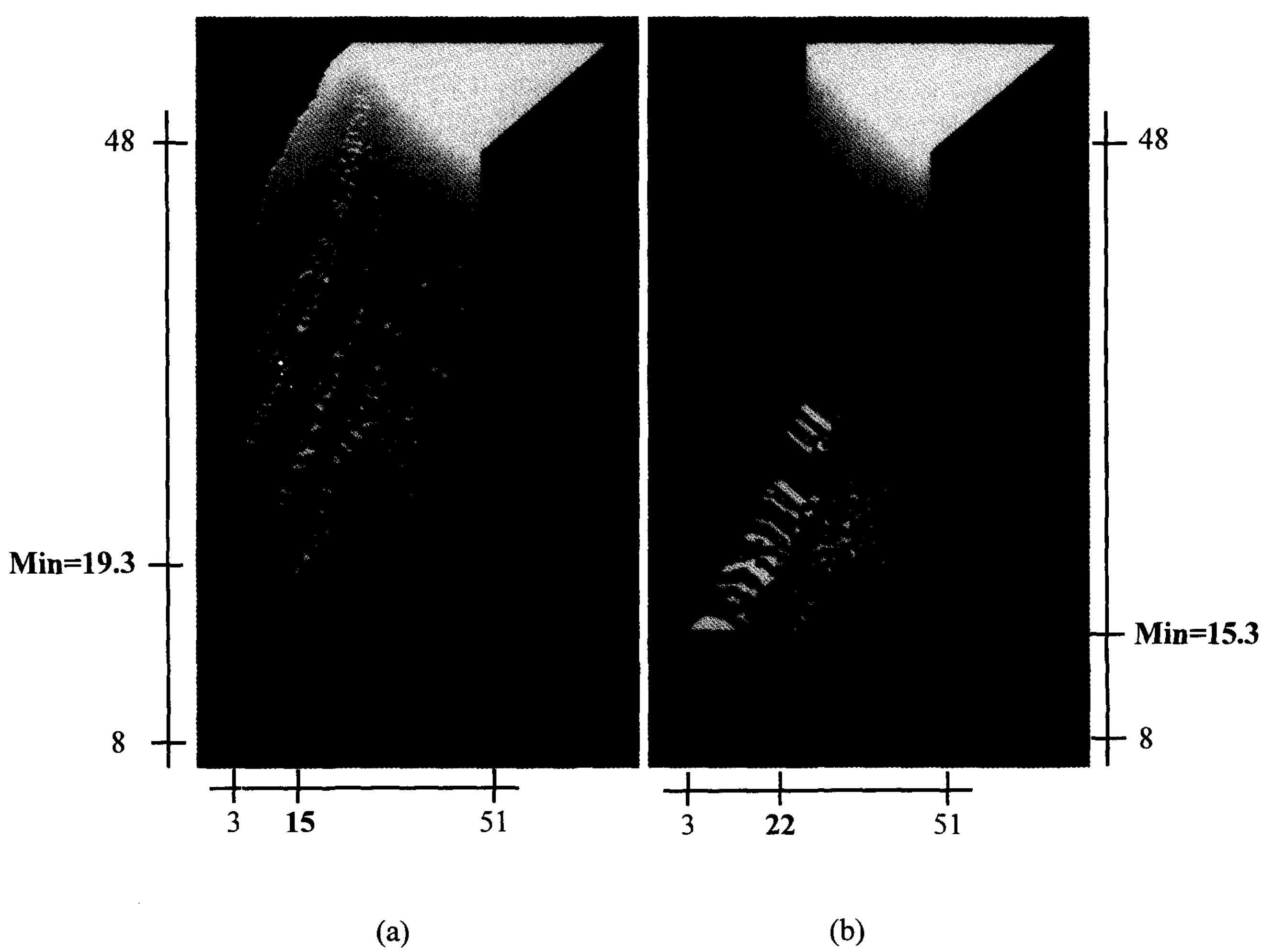


Figure 6 Representation of cost values with the methods of: (a) Reilly, Cooper and Elbaum; and (b) K nearest neighbours with $k = 2$

The total number of features of each defect is 51, for the two classification methods, three features were imposed at the starting point of the algorithm of features selection. So, the number of classifiers built during the iterative process is 1176 for each classification method (see Equation (5)). These cost values are represented as a 3D image for the RCE and KNN methods in Figure 6, by using the principle described in Figure 5.

The algorithm to find the best feature combination shows that the best classifier is obtained with 15 features for the method of Reilly, Cooper and Elbaum (a) for a cost of 19.3 and 84.3% well classified defects, and 22 features for the K nearest neighbours method (b) for a minimal cost of 15.3 and 79.3% well classified defects. The comparison between these two methods is easier and more interesting with the images of the cost values. The representation of the KNN shows that the cost value decreases more rapidly than with the RCE. Moreover, cost values seem to be more stable around a number of 10 features. On the two representations, the cost values are more chaotic with more than 30 features.

We should choose the KNN classification method with 10 features corresponding to a cost of 16. It seems to be the area where the robustness is the better because of the stability of cost values. We have tested and verified this robustness with different test sets.

To evaluate the benefit of our method, we have calculated the cost obtained with the classification giving the best rate of well classified defects. The cost value is 30% higher than the cost value obtained with the cost optimization.

Care should be taken about some conclusions concerning the performance between the different classification methods. In our application, the database has extremely varying class sizes and their statistical representativity is not good but corresponds to the true occurrence of the defects. The comparison is only valid in our application context and shows that the cost matrix optimization can be applied successfully on two different classification methods.

Conclusions

In this paper, we have developed an optimization method based on the writing of a cost matrix. This is used for a recognition system and allows an evaluation adapted to each application. We describe a method in order to write this cost matrix by taking into account the specificities of the application. The method has been applied to a real case of defect identification on flat steel products and the results obtained show a significant improvement in the performance of the identification system.

Moreover, we propose a representation technique of the criterion measured during the optimization stage. This image should allow us to choose the best classifier and to appreciate its robustness.

At the present time, we are working on the relationship between the representation and the real robustness. The results will be presented in future publications.

Acknowledgements

This research is financially supported by the European Community of Carbon and Steel (CECA) under contract no. 7210 GB/311.

References

1 **Ho, M.** 'Surface inspection system with defect classification', *Iron Steel Eng* (May 1990) pp 26–29
2 **Porter, T.** 'On-line inspection of surface defects', *Steel Tech Int* (1988) pp 334–336
3 **Rautaruukki** 'Defect classification in surface inspection of strip steel', *Steel Times* (May 1992) pp 214–216
4 **Canella, G. and Falessi, R.** 'Surface inspection and classification plant for stainless steel strip', *Non-Destructive Testing*, Sao-Paulo (1992) pp 1185–1190
5 **Seitzler, T.** 'Industrial vision tackles web inspection', *Adv Imag* (February 1993) pp 32–34
6 **Dupont, F.** 'Analyse d'image et reconnaissance des formes: application à l'identification automatique de défauts de surface sur tôles décapées', Ph.D. thesis, INSA Lyon (May 1994)
7 **Ferriere, D.** 'Contrôle par voie optique des brames de coulée continue', CECA Contract Number 7210 GB 302 (March 1987)
8 **Fukunaga, K.** 'Estimation of classifier performance', *IEEE Trans Patt Anal Mach Intell* **11** 10 (1989) pp 1087–1101
9 **Vallet, F.** 'Approche globale du problème de discrimination – aspects probabilistes', *Rev Techn THOMSON-CSF* **22** 4 (1990) pp 519–541
10 **Jender, H.** 'Contrôle temps réel par vision artificielle de tubes métalliques en défilement continu', Ph.D. thesis, Université de Bourgogne (1989)
11 **Celeux, G.** 'Analyse discriminante sur variables continues', INRIA, Collection didactique, Rocquencourt (1990)
12 **Reilly, D. L., Cooper, L. N. and Elbaum, C.** 'A neural model for category learning', *Biol Cybern* **45** (1982) pp 35–41
13 **Fukunaga, K.** *Introduction to Statistical Pattern Recognition* Academic Press, New York (1972)
14 **Hong Y.** 'Prototype optimization for nearest neighbor classifiers using a two layer perceptron', *Patt Recogn* **26** 2 (1993) pp 317–324