The Third AFSS(1998). 208-212

DESIGN OF A BINARY DECISION TREE FOR RECOGNITION OF THE DEFECT PATTERNS OF COLD MILL STRIP USING GENETIC ALGORITHM

Byung Jin LEE[a] , Kyoung LYOU , Gwi Tae PARK, Kyoung Min KIM[b]

a. Dept. of Electrical Engineering, Korea University
5-Ga, Anamdong, Sungbuckgu, Seoul, 136-701 Korea
Tel: +82-2-927-3268 Fax: +82-2-921-0163 E-mail: jelux@eeserver.korea.ac.kr
b. Dept of Electrical Engineering, Yosu National University
195, Kuk-dong, Yosu-shi, Chollanam-do, 550-749 Korea

Abstract

This paper suggests the method to recognize the various defect patterns of cold mill strip using binary decision tree constructed by genetic algorithm automatically. In case of classifying the complex patterns with high similarity like the defect patterns of cold mill strip, the selection of the optimal feature set and the structure of recognizer is important for high recognition rate. In this paper genetic algorithm is used to select a subset of the suitable features at each node in binary decision tree. The feature subset of maximum fitness is chosen and the patterns are classified into two classes by linear decision function. After this process is repeated at each node until all the patterns are classified respectively into individual classes. In this way, binary decision tree classifier is constructed automatically. After constructing binary decision tree, the final recognizer is accomplished by the learning process of neural network using a set of standard patterns at each node. In this paper, binary decision tree classifier is applied to recognition of the defect patterns of cold mill strip and the experimental results are given to show the usefulness of the proposed scheme.

keywords: genetic algorithm, decision tree, feature selection, defect pattern, cold mill strip

1. Introduction

To produce cold mill strip of high quality, it is important to extract the defects on the surface of cold mill strip rapidly in manufacturing process. So, efficient methods for extraction and recognition of the defect patterns of cold mill strip have been studied. Recently, to substitute defect-extraction system using one dimensional reflected laser signal, the pattern recognition method by image processing have been tried

The conventional method to recognize the defect patterns using image processing is to extract good features experimentally after preprocessing the image acquired from CCD camera, , and to recognize the patterns in one step by inputting all the features to neural network. But the conventional method has two problems when the characteristics of the defect patterns are considered. At first, because the shapes of the defect patterns are complex and have low regularity, the recognition rate of defect patterns is sensitive to the kinds of the selected features, and because even though the features have good separability respectively, they may disturb the separability one another when they are used together, if the features are selected experimentally, the fitness of the selected feature subset can not be guaranteed. At second, because there exist some similar classes of defect patterns which can be classified into a group, to classify all the patterns in only one step causes the high classification error.

To solve these problems, we suggest the multi-stage classifier like decision tree, which repeats decisions so as to classify patterns individually. The decision tree classifier makes a decision fast and exactly by dividing the complex and global decision into several simple and local decisions[6].

For the efficient and accurate classification, the optimal or nearly optimal feature subset within the feature space need to be selected at each decision node[3]. There is a three-fold potential advantage in applying a method of selecting subset of features for input to the classification. Firstly the performance of the classifier can be improved by reducing the possible inputs to a set of relevant uncorrelated variables. Secondly the use of a smaller set of features not only reduces the time complexity, but also reduces the processing time needed to produce the feature set, thus speeding up the response time of the system. Finally since there is a direct relationship between the dimensionality of a problem and the size of the example set needed to adequately cover the problem space, the reduction of the feature set means that smaller off-line training set can be used with all the ensuring benefits in terms of data-collection and training times.

In this paper, genetic algorithm is used to find a subset that yields the lowest error rate of a classifier. The advantage of this search method is that it will tend to produce nearly optimal solution quickly. The fitness function used to evaluate the selected subsets of features is the error estimator for the linear decision function which is also produced by genetic algorithm. Genetic algorithm makes a linear decision function of which the dimension is that of the selected feature space and searches a linear decision function which minimize the classification error. With this error, the fitness of the selected feature subset is calculated. Finally, the feature subset with maximum fitness is chosen and the patterns are classified into two classes by the linear decision function. This process is repeated at each node until all patterns are classified respectively into individual classes. In this way, the binary decision tree classifier is constructed automatically.

After constructing binary decision tree, the final recognizer is accomplished by the learning process of neural network with a set of standard patterns at each node.

In this paper, first of all, we introduce binary decision tree and describe the method of generating linear decision function using genetic algorithm and the method of selecting a feature subset also using genetic algorithm. Then, we describe the method of constructing binary decision tree and finally, we apply the proposed classifier to recognizing defect patterns of cold mill strip and show experimental results.

2. Construction of binary decision tree using genetic algorithm

2.1 Binary decision tree

The classifier which classifies input patterns in only one step using the whole features or a part of them, is called 'one-stage classifier'. In case of one-stage classifier, it can not be guaranteed that the whole features used in classification have the best separability, and the efficiency of the classifier is low because it compares a pattern with all of the other classes. To solve these problems of one-stage classifier, the classifier that decide the class of the input pattern by repeating two or more decisions successively, is designed and it is called 'multi-stage classifier' or 'decision tree classifier'.

The advantages of the decision tree are as follow.

i) It approximates the complex global decision to the union of the simpler local decisions.

ii) One-stage classifier compares the input pattern with the patterns of all classes. But because the binary decision tree compares the input pattern with the only patterns of partial classes, unnecessary computations can be reduced.

iii) It improves the recognition rate, because it can select an optimal feature set that can maximize the discriminant ability at each node of the tree.

It is called 'binary decision tree classifier' that has two child nodes at each node. Binary decision tree classifier classifies the patterns into two classes with a suitable feature subset at each node, and this process is iterated until there exists only one pattern class in each leaf node.

2.2 Feature selection using genetic algorithm

2.2.1 Feature selection problem

If the only necessary features are used at each node of binary decision tree classifier, the accuracy and the credibility of the classifier are increased. So to select a valid feature subset in the entire feature set is needed, which is called 'the feature selection problem'.

To select the optimized feature subset, the separability of all the combinations of the features should be evaluated. However, the number of combinations that m components are selected among n component is expressed as nCm, which is a large value even though n and m are not large.

There are some searching algorithms to avoid the exhaustive searching like above, which are top-down, bottom-up, branch and bound, and so on. The feature selection problem can be regarded as an optimization problem. So in this paper, the feature selection is executed using genetic algorithm which is a global optimization technique.

2.2.2 feature selection using genetic algorithm

Because genetic algorithm has higher probability to search the global optimized solution than other classical optimization algorithm by searching the multiple global solutions at a time, the optimal feature subset can be selected effectively[2].

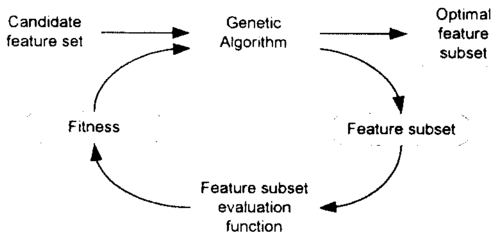


Fig. 1 Processing block diagram of feature selection

Fig. 1 is the block diagram which shows the process that the optimal feature subset is selected by genetic algorithm. In genetic algorithm, the chromosomes represent the feature subsets, and the fitness of each chromosome is calculated by evaluating the validity of the feature subset, and the survival probability is determined. According to the probability, the operations of evolution are executed. In this way, the new evolved feature subsets are generated. The optimal feature subset is acquired by iterating the evolving process. The process of evaluating a feature subset is the most important process to get the optimal solution. In this paper, we evaluate a feature subset by the classification error when classifying patterns with the linear decision function which is also generated by genetic algorithm, which will be described in next chapter.

Suppose that there exists an n-bit binary string $a = \{a_1, \ldots, a_n\}$ which represents a feature subset, where the number of all features is n. If a_i is 1, it means that the i-th feature is selected, inversely if a_i is 0, it means that the i-th feature is not selected. The feature selection algorithm is described below by this assumption.

i) Initialize the string set.

ii) Transform each string into a feature subset

iii) Calculate classification error and evaluate the feature subset and calculate the fitness of the string using the fitness function.

iv) If there exists a chromosome that reach to the desired fitness, finish algorithm.

v) With the fitness of each chromosome, execute the basic genetic operations, which are reproduction, crossover and mutation.

vi) If it reach to the maximum pre-defined number of generations, finish the algorithm and select the string which has the best fitness as a final result. If not, go to step ii).

2.3 Evaluation a feature subset

2.3.1 Determination the linear decision function using genetic algorithm

The following method is used to minimize the classification error using genetic algorithm.

Supposing that given data set is $X=\{x_1, x_2 .. x_N\}$ ($x_k \in R^n$, n is the number of features) , We can define $l(j)$ as minimum value and $r(j)$ as maximum value that include the whole value of j-th feature.

$$l(j) = \min_i x_{ij}$$
$$r(j) = \max_i x_{ij} \tag{1}$$

In case of 2-dimensional space, j can have the value of 1 or 2 and on the basis of $l(j)$ and $r(j)$, we can imagine a rectangle that can includes whole data. Inside of the rectangle, we can select two points arbitrarily, and find a line that passes the points chosen previously. From the coefficients of the line function, we can get w_1, w_2, w_3 of 2-dimensional decision function in Eq. (2).

$$d(\boldsymbol{x}) = w_1x_1 + w_2x_2 + w_3 = 0 \tag{2}$$

If we expand 2-dimensional case to n-dimensional case, we can find a hyperplane by selecting n-points in the hyperspace. And we can also find the values w_1, w_2, .. , w_{n+1} which appear in linear decision function of n-dimensional space, Eq. (3).

$$\begin{aligned} d(\boldsymbol{x}) &= w_1x_1 + w_2x_2 + \cdots + w_nx_n + w_{n+1} \\ &= \boldsymbol{w}_0{}' \boldsymbol{x} + w_{n+1} \end{aligned} \tag{3}$$

When we match this concept with a binary string of genetic algorithm, n-segments of a binary string indicate one point in n-dimensional space. In n-dimensional case, we should select n-points, so a string is composed of n^2 -segments. Supposing that the length of each segment is m-bit, the total length of a binary string becomes n^2m -bit.

Let s_j be the decoded values of j-th segment. By Eq. (4), we can match the smallest value of s_j, '0' with $l(j')$, the largest value of s_j, (2^m-1) with $r(j')$. Then we can convert the value of s_j to the value d_j which exist in the interval $[\,l(j'), r(j')]$, and an n^2m-bit binary string indicates n-points in the minimum sized hypercube that can include all of the given data.

$$d_j = \frac{r(j') - l(j')}{2^m - 1} \cdot s_j + l(j') \tag{4}$$

where $j' = (rem(j-1, n+1)+1)$, $rem(a, b)$ is the remainder when we divide a by b.

The algorithm that determines the weights of linear decision function using genetic algorithm is as follows.

i) Initialize binary string set.

ii) Convert each binary string into a point.

iii) By using the method described above, find the weight of hyperplane that consists of the points.

iv) Calculate the classification error using the sample patterns by the decision function which is determined from the weights and calculate fitness.

v) If there exists a chromosome that reach to desired fitness, finish algorithm.

vi) With the fitness of each chromosome, execute the basic genetic operations, which are reproduction, crossover and mutation.

vii) If it reach to the maximum pre-defined number of generation, finish the algorithm and select the string which has the best fitness as a final result. If not, go to step ii).

2.3.2 Evaluation of a feature subset

The decision function described above is the linear decision function that minimizes the error which occurs during classifying standard patterns into two classes in certain feature space. However, the real input patterns are not restricted to two classes of patterns, we define the classification error as the number of patterns which exist in different region when we classify several classes of patterns into two groups. Genetic algorithm determines the decision function that minimize classification error in given feature space. The minimized error varies with the combination of features. Hence the fitness function is constructed to give high fitness for a combination with small classification error, and low fitness for a combination with large classification error.

2.4 Construction of the binary decision tree and the final recognizer

Using the method described above, we choose certain feature subset which minimizes classification error, and we classify patterns into two groups for each node with this feature subset. Binary decision tree is constructed by iterating this process until all the classes of patterns appear independently at each leaf node.

Because we construct binary decision tree not for two classes, but for multiple classes, it is better to maintain uniform distribution for two separated groups at each node, which means it is better that two separated groups have the similar number of classes without partiality. To quantify this, We define the balance coefficient using the mean of classes and the deviation of classes of a new group, like Eq. (5). If the number of patterns of two separated groups are similar, the balance coefficients would be smaller. In this case, because the depth of the binary tree becomes small, the matching times for recognizing a pattern decrease. The smallest value of balance coefficient is '0', the largest value is '$\sqrt{2}$' for binary tree case.

$$balance = \sqrt{\frac{\sum_{j=1}^{h}(N_j - \frac{N}{h})^2}{(\frac{N}{h})^2}} \tag{5}$$

At Eq. (5), h means the number of nodes, N means the number of the input patterns, N_j means the number of the patterns which included j-th node. In this paper, we construct binary tree, so h becomes '2'. the Fitness function that includes balance coefficient is Eq. (6).

$$fitness = \frac{1}{1 + w_e \cdot error + w_b \cdot balance} \tag{6}$$

At Eq. (6), *error* means classification error, *balance* means the balance coefficient between groups. And w_e, w_b mean the weights for weighting each parameters. If both the classification error and the balance coefficient have the value '0', fitness has the largest value '1'. And the result of the constructed tree can be varied with the adjustment of the weights w_e, w_b. For example if we assign large value to w_b, the probability which we can get more balanced tree structure becomes high, while the error rare becomes high.

After construction of binary decision tree, by training BP neural network with the feature subset selected optimally at each node, the final binary tree structured recognizer is accomplished

3. Classification of the defects of cold mill strip using binary tree classifier

3.1 Extraction of the features of the defect pattern

The types of the defect patterns of cold mill strip can be classified into about seven classes, which are Dull, Oil-drop, Slip, Dent, Scale, Dirt, and Scratch. After preprocessing the acquired image, we extract six candidate features as follow.

3.1.1 Geometrical features

In this paper, we select the geometrical features, such as the area, the area ratio, and compactness as candidate features, which are not related to the size and the direction of the patterns.

① def_area : the area of a pattern
(the number of pixels of a defect pattern)
② area_rat : the ratio of def_area to box_area
(area_ratio = def_area / box_area)
box_area is the area of the smallest rectangle which includes defect pattern inside
③ compactness : compactness of a pattern
($(4\pi$ *area)/perimeter^2)
where perimeter is the length of outline.

3.1.2 Feature of moment.

The concept of moment which is defined in the theory of probability is used widely in pattern recognition as a practical method to extract the features of the shape [9]. Among the features of moment, the informations which are useful for the inspection of the defects of cold mill strip are the length of the longest axis, the ratio of longest axis to shortest axis, and spread of a pattern. These features can be calculated by Eq. (7)~(11).

① The length of longest axis and the shortest axis of a pattern

$$\mu_{ij} = \sum_x \sum_y (x-\bar{x})^i \; (y-\bar{y})^j f(x,y) \tag{7}$$

$$a = 2\sqrt{2}\sqrt{(\mu_{20} + \mu_{02} + \sqrt{((\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2)})}\text{:longest} \tag{8}$$

$$b = 2\sqrt{2}\sqrt{(\mu_{20} + \mu_{02} - \sqrt{((\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2)})}\text{:shortest} \tag{9}$$

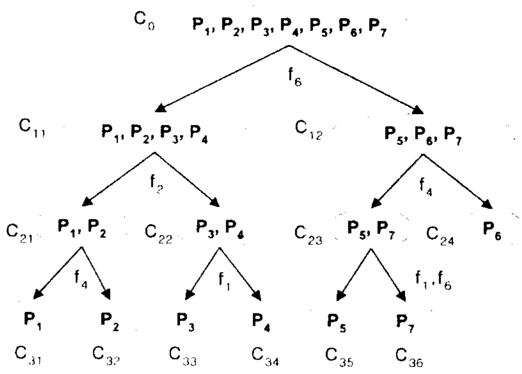
where $f(x,y)$ is the function of gray level of an image, and μ_{ij} is the central moment of a pattern.

② The ratio of longest axis to shortest axis of a pattern

$$axis\_ratio = \frac{b}{a} \tag{10}$$

③ Spread

$$spread = \frac{\frac{\mu_{02} + \mu_{20}}{2}}{\mu_{00}^2} \tag{11}$$



symbol	P_1	P_2	P_3	P_4	P_5	P_6	P_7
pattern	dull	oil	slip	dent	scale	dirt	scratch

symbol	f_1	f_2	f_3	f_4	f_5	f_6
feature	def_area	area ratio	compact	axis_ratio	spread	long_axis

Fig. 2 Binary decision tree constructed with standard patterns

3.2 Construction of the binary tree structured recognizer

The data used in constructing the binary tree structured recognizer are the feature vectors extracted from seven kinds of standard defect patterns. In construction of binary tree using genetic algorithm, we set the weights, w_e and w_b 1 in Eq. (6). Fig. 2 shows the binary decision tree constructed with standard patterns by genetic algorithm. In Fig. 2, P_i is a sort of pattern, f_i is a feature, and C_{mn} represents a class at each node. Table 1 shows the classification error, the balance coefficients, and the fitness values at each node. The classification error at Table. 1 is the number of patterns which leave their class when the patterns are divided into two groups at the node. Table 2 shows the weights of linear decision function at each node.

Table 1 The patterns and fitness at each node

Node	Pattern	Error/ patterns	Feature	Fitness
C_0	P_1 P_2 P_3 P_4 P_5 P_6 P_7	0/38	f_6	0.7180
C_{11}	P_1 P_2 P_3 P_4	0/15	f_2	0.5677
C_{12}	P_5 P_6 P_7	2/23	f_4	0.2550
C_{21}	P_1 P_2	0/8	f_4	0.5858
C_{22}	P_3 P_4	0/7	f_1	0.7795
C_{23}	P_5 P_7	3/21	f_1 f_6	0.1874

Table 2 Decision function at each node

$d(x)$	w_1	w_2	w_4	w_3	w_5	w_6	w_7
$d_0(x)$	0.000	0.000	0.000	0.000	0.000	-1.765	1.000
$d_{11}(x)$	0.000	-2.326	0.000	0.000	0.000	0.000	1.000
$d_{12}(x)$	0.000	0.000	0.000	-1.853	0.000	0.000	1.000
$d_{21}(x)$	-1.713	0.000	0.000	0.000	0.000	0.000	1.000
$d_{22}(x)$	0.000	0.000	0.000	-2.304	0.000	0.000	1.000
$d_{23}(x)$	-4.062	0.000	0.000	0.000	0.000	1.5902	1.000
$d(x) = w_1x_1 + w_3x_3 + w_4x_4 + w_5x_5 + w_6x_6 + w_7 = 0$							

At each node constructed above, the final recognizer is accomplished by training the BP neural network with the selected feature subset. The number of nodes in input layer is set to the number of the selected features, and the number of nodes in hidden layer is set 10. By setting the number of nodes in output layer 2, we make the output layer represent the binary decision.

Table 3 shows the results of recognizing the defect patterns of cold mill strip using the binary tree structured recognizer

Table 3 Recognition rate of each defect pattern

recognition / pattern	recognition rate (recognition/pattern)	recognition rate(%)
Dent	0/3	0%
Dull	6/12	50%
Oil_drop	4/4	100%
Slip	1/4	25%
Dirt	2/2	100%
Scale	19/22	86.3%
Scratch	7/8	87.5%
Total	39/55	71%

At Table 3, the recognition rates of Dent and Slip are very low. However Table 1 shows that the linear classification errors are zero at nodes C_0, C_{11}, C_{22} when constructing binary decision tree. this means that the standard patterns of Dent and Slip are classified linearly. Because the least number of features which fit to classify standard patterns are selected, if the number of standard patterns is small, the recognizer becomes sensitive to noise.

4. Conclusion

In this paper, we used binary decision tree structured classifier to recognize defect patterns of cold mill strip. At each node of binary tree, genetic algorithm, which is one of the powerful optimization algorithm, was used for the selection of the best feature subset, and linear decision function which was also generated by genetic algorithm, was used to calculate the fitness of a feature subset. There are two advantages of this method. The one is that the construction of binary decision tree and the selection of the best feature subset can be executed automatically for the given patterns. The other is that by designing the fitness function of genetic algorithm properly, we can get the decision tree with the consideration of balance of the classes as well as the classification error.

References

1. G. P. Babu and M. N. Murty, "Clustering with evolution strategies", *Pattern Recognition*, vol. 27, pp. 321-329, 1994.
2. S. W. Jung and G. T. Park, "Design and Application of Binary Decision Tree using Genetic Algorithm", *Journal of the KITE*, vol 33, No. 6, pp. 1122-1130, 1996.
3. F. Brill, D. Brown & W. Martin "Fast Genetic Selection of Features for Neural Network Classifiers", *IEEE Trans.* on Neural Networks vol. 32, pp. 324-328 March 1992.
4. L. Yao, "Nonparametric learning of decision regions via the genetic algorithm", *IEEE Trans. Sys., Man, Cybern.*, vol. 26, pp. 313-321, Apr. 1996.
5. J. K. Mui and K. S. Fu, "Automated classification of nucleated blood cells using a binary tree classifier", *IEEE Trans. Patt. Anal. Mach. Intl.*, vol. 5, pp. 429-443, Sep. 1980.
6. S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology", *IEEE Trans. Sys., Man, Cybern.*, vol. 21, pp. 660-674, May/Jun. 1991.
7. H. J. Payne and W. S. Meisel, "An algorithm for constructing optimal binary decision trees", *IEEE Trans. Computers*, vol. 26, pp. 905-916, Sep. 1977.
8. P. H. Swain and H. Hauska, "The decision tree classifier: design and potential", *IEEE Trans. Geosci. Elec.*, vol. 15, pp. 142-147, Jul. 1977.
9. M. D. Levine, "Feature Extraction : A Survey", *Proceedings of the IEEE* , pp. 1391-1407, 1969