Research Paper

Volume : 3 | Issue : 5 | May 2014 • ISSN No 2277 - 8179

Automatic Edge/Fault Detection Using Hybrid Technology of GA and MLP of Image Processing: A Review

Engineering

KEYWORDS : Generic algorithm, multilayer perceptron, cracks detection, cracked tiles, hybrid

Mehul M. Baraiya	PG Research Scholar, Department of Electronics & Communication, RK University, Rajkot(GUJ), India.
Dhaval S. Pipalia	Assistant Professor, Dept. of Elect. & Comm., RK University, Rajkot, India.

ABSTRACT

In tiles industries, the problem of the crack can be avoided by detecting the cracks on the surface of tiles. The efficient use of image processing with advanced technologies Generic Algorithm and Multilayer Perceptron with self organizing maps which would be required in the real time applications of tiles surface crack detection. Till now it was possible with individual above mentioned methods. This paper shows the possibility of hybrid method combining GA and MLP. Crack/ Fault detection using hybrid of GA and MLP will be helpful for deduction in the rejection ratio of the tiles at the time of packaging.

1. INTRODUCTION

It is compared that three methods for automatically classifying pavement cracks, genetic algorithm, multilayer perceptron, and self organizing maps. The best classifier demonstrates accuracies between 86 to 98 %. It can be improved if any two techniques combined. (Hough transforms and projection methods will be used.) So that by hybrid techniques it can be improve up from 90 to 100 % in surface crack/fault detection.

Tiles and Concrete structure are usually constructed with material that exhibits distress over time due to loading, environment conditions and normal wear. Often the distresses are present in the form of surface cracking. [3]

A multilayer perceptron (MLP) is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable.

Artificial intelligence a generic algorithm is a search heuristic that mimics the process of natural selection. This heuristic is routinely used to generate useful solutions to optimization and search problems.[7] Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection and crossover. Genetic algorithms find application in bioinformatics, physics, pharmcometrics and other fields.

There are main four types of crack which are focused in this review paper. Since a large part of our tiles manufacturing infrastructure, a number of distresses have been identified and their characteristics cataloged [5]. Four common crack types are illustrated in Figure 1, and form the target of our classification system.

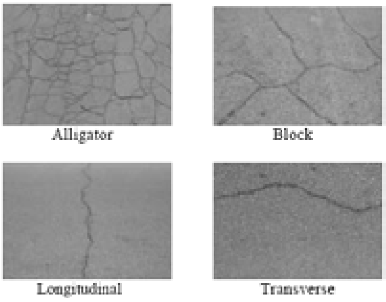

Figure 1: Types of pavement cracks. [1]

2. GENETIC ALGORITHM

In a genetic algorithm, a population of candidate solutions called individuals, creatures, or phenotypes to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and is an iterative process, with the population in iteration called a generation. In each generation, the fitness of every individual in the population is evaluated; the fitness is usually the value of the objective function in the optimization problem being solved. The more fit individuals are stochastically selected from the current population, and each individual's genome is modified (recombined and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

A typical genetic algorithm requires [1]:

- A genetic representation of the solution domain,
- A fitness function to evaluate the solution domain.

A standard representation of each candidate solution is as an array of bits.[2] Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in genetic programming and graph-form representations are explored in evolutionary programming; a mix of both linear chromosomes and trees is explored in gene expression programming. Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators. Diversity refers to the average distance between individuals in a population. A population has high diversity if the average distance is large; otherwise it has low diversity. In the following figure, the population on the left has high diversity, while the population on the right has low diversity.

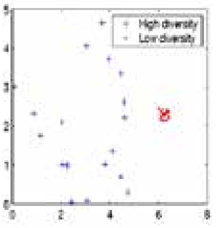

Figure 2: GA extraction from Matlab

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, allowing the entire range of possible solutions. Occasionally, the solutions may be “seeded” in areas where optimal solutions are likely to be found.

3. MULTILAYER PERCEPTRON

Simple perceptron, which involves feed-forward learning based on two layers: inputs and outputs. A little more complexity is done by including a third layer, or a hidden layer into the network. A reason for doing so is based on the concept of linear separability. While logic gates like “OR”, “AND” or “NAND” can have 0′s and 1′s separated by a single line or hyperplane in multiple dimensions, this linear separation is not possible for “XOR” (exclusive OR)

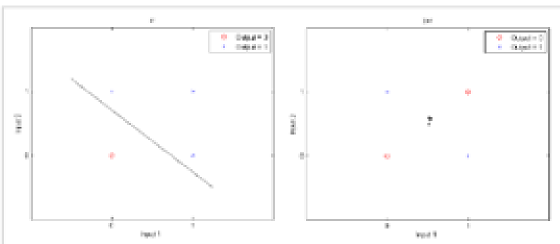

Figure 3: AND and OR extraction of MLP from Matlab

A multilayer perceptron (MLP) is a feed forward artificial neural network model that maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable. If a multilayer perceptron has a linear activation function in all neurons, that is, a simple on-off mechanism to determine whether or not a neuron fires, then it is easily proved with linear algebra that any number of layers can be reduced to the standard two-layer input-output model. What makes a multilayer perceptron different is that each neuron uses a nonlinear activation function which was developed to model the frequency of action potentials, or firing, of biological neurons in the brain. This function is modeled in several ways, but must always be normalize and differentiable.

The two main activation functions used in current applications are both sigmoid, and are described by

In which the former function is a hyperbolic tangent which ranges from -1 to 1, and the latter, the logistic function, is similar in shape but ranges from 0 to 1. Here yi is the output of the ith node (neuron) and is the weighted sum of the input synapses. More specialized activation functions include radial basis functions which are used in another class of supervised neural network models.

Layers

The multilayer perceptron consists of three or more layers (an input and an output layer with one or more hidden layers) of nonlinearly-activating nodes. Each node in one layer connects with a certain weight ij to every node in the following layer. Some people do not include the input layer when counting the number of layers and there is disagreement about whether ij should be interpreted as the weight from i to j or the other way around[1].

Back Propagation

Learning occurs in the perceptron by changing connection weights after each piece of data is processed, based on the amount of error in the output compared to the expected result. This is an example of supervised learning, and is carried out through back propagation, a generalization of the least mean squares algorithm in the linear perceptron. We represent the error in output node j in the nth data point by

$e_j(n) = d_j(n) - y_i(n)$

Where d is the target value and y is the value produced by the perceptron. We then make corrections to the weights of the nodes based on those corrections which minimize the error in the entire output, given by

$$\varepsilon(n) = \frac{1}{2}\sum_j e^2{}_j(n)$$

Applications of MLP

MLPs were a popular machine learning solution in the 1980s, finding applications in diverse fields such as speech recognition, image recognition and machine translation software, but have since the 1990s faced strong competition from the much simpler support vector machines. More recently, there has been some renewed interest in back propagation networks due to the successes of deep learning [6].

Multilayer perceptron using a back propagation algorithm are the standard algorithm for any supervised learning pattern recognition process and the subject of ongoing research in computational neuroscience and parallel distributed processing. They are useful in research in terms of their ability to solve problems stochastically, which often allows one to get approximate solutions for extremely complex problems like fitness approximation.

4. COMPARATIVE ANALYSIS

The MLP consisted of two input units, one output unit for each of the four possible crack types, along with a variable number of hidden units. The number of hidden units was assessed empirically with both the Hough and projection representations, resulting in three hidden units being chosen for this study.

The genetic classification by evolving a classification matrix that would map the target classes into distinct regions on a matrix that could then be indexed by feature coordinates from the testing set. A typical evolved GA matrix is illustrated.

In addition to providing a similar output representation as the genetic approach described above, the SOM is an example of an unsupervised neural network. The same method for classification and resolution used in GA was used in SOM.[1]

Table 1: Comparative Analysis of MLP and GA [1]

PARAMETER	MLP	GA	COMBINATION
Complexity	Less	More	More
Power	More	Less	Less
Accuracy	More	Less	More
Accuracy in %	94 %	92.8 %	>95%
Information Loss	More	Less	Less
Implementation	Easy	Not Easy	Not Easy

5. CONCLUSION

After all the review, this paper comes to the conclusion that if these two methods combined then the efficiency of the crack detection can be improved. This system, in addition to producing competitive accuracy, has the positive attributes of design simplicity and computational efficiency. New efficient algorithms can be developed to identify the crack/fault and the depth of the crack. Using the hybrid algorithms some parameters can be extended and system can be made of high performance.

6. ACKNOWLEDGEMENT
Our thanks to the experts who have contributed towards development of the template. Hearty thanks are expressed to Prof. A. A. Bavarva (Electronics & Communication Department, RK University) for encouraging and proving necessary help at each level. We would like to thank Prof. D. U. Shah (Head of Electronics & Communication Engineering Department, RK University) for his valuable suggestions and timely guidance throughout our work and enthuse us by giving all type of facility.

REFERENCE

H. Rababaah, D. Vrajitoru, J. Wolfer, "Asphalt Pavement Crack Classification: A Comparison of GA, MLP and SOM", Computer and Information Sciences, Indian University South Bend, 1700 Mishawak Avenue. | | [2] S. T. Saranam, S. Chatrati ,R. Yakkala, "An Approach to Improve the Railway Crack Detection in the Tracks by Automated Engine.", IOSR-JECE , Vol.6 ,Issue – 6 , p-ISSN: 2278-8735 , Aug-2013. | | [3] Timm and McQueen, "A study of manual vs. automated pavement condition surveys", Auburn Alabama, 2004, http://www.eng.auburn.edu. | | [4] Mclean, "Distress Identification Manual", Federal highway Administration, US-Department of Transportation, FHWA-03-031, 2003, http://www.tfhrc.gov. | | [5] Mohajeri, Manning, "An Operating System of Pavement Distress Diagnosis by Image Processing", Transportation Research Record 1311, TRB, National Research Council, Washington, D.C., pp. 120-130, 1991. | | [6] N. Senthilkumar, R. Rajesh, " Edge detection techniques for Image Segmentation – A survey of soft computing Approaches", International Journal of recent trends in engineering, Vol. 1, No. 2, May 2009. | | [7] F. M. Ham, I Kostnic, "Principal of Neurocomputing for Science & Engineering", McGraw Hill publications, 2001. | | [8] G. Cybenko, "Approximation by superposition of sigmoidal function Mathematics of Control, Signal and Systems", 2(4), 303-314, 1989. | | [8] R. C. Gonzalez, R. E. Woods, S. L. Eddins, "Digital Image Processing using MATLAB", ISBN 81-297-0515-X, 2005, pp. 76-104,142-166. |