A neuro-fuzzy approach for automatic classification of wooden plates

Adilson Gonzaga\textsuperscript{a}, Celso A. França\textsuperscript{b}, Annie F. F. Slaets\textsuperscript{b}

\textsuperscript{a}Electrical Engineering Department, School of Engineering at São Carlos
\textsuperscript{b}Physics Institute of São Carlos, University of São Paulo, Brazil

ABSTRACT
This paper discusses a visual inspection methodology applied on plates of wood used in pencil industry. A Neuro-Fuzzy approach was established taking into account the visual characteristics of nodes, stripes and surface defects. An Artificial Neural Network based on MIN and MAX neurons, was implemented with fuzzy rules and fuzzy input variables. The ANN has four neurons in the input layer according to the features: the range width of pixel distribution, the elbow region of the histogram, the global contrast and the stripe quantity. Three hidden layers do the fuzzyfication, the MIN rule and the MAX rule. One output neuron with a sigmoidal function does the defuzzyfication and classifies the plates. Three classes are obtained: the A class represents the best plates, the S class is drawn up by the worst plates and the C class depends on demand. The experimental results show that it’s possible to classify about 153 plates/min, within three classes, at a reliability of 90% using a low cost equipment in an industrial plant.

Keywords: pattern recognition, computer vision, neural networks, fuzzy logic, neuro-fuzzy system, classifier patterns, image processing, backpropagation neural network, OR/AND neuron, supervised learning.

1 - INTRODUCTION
Pattern recognition implies a wide range of information processing problems with great practical significance. These problems range from speech recognition and handwritten character classification to machinery fault detection and medical diagnosis\textsuperscript{1}. A pattern recognition system can be considered as a two stage device\textsuperscript{2}. The first one is the feature extraction and the second one is the classification. Extracting good features is the essence of pattern recognition\textsuperscript{3}. A bad choice of features can not separate different classes.

A lot of scientific effort has already been dedicated to pattern recognition problems, especially classification procedures\textsuperscript{4}. Neural techniques are being researched for implementing classifiers that minimize the classification error rates, optimize training and classification time, and are adaptive and memory efficient\textsuperscript{5}. The human brain is excellent at performing many of the tasks that would be desirable in artificial systems, such as vision, speech recognition, learning by example and so on\textsuperscript{6}. Images of real scenes very frequently contain data that is incomplete and ambiguous\textsuperscript{6}. The utility of fuzzy sets lies in their ability to model the uncertain or ambiguous data so often encountered in real life\textsuperscript{7}. Therefore, to enable an artificial system to tackle real-life situations, like humans, one may incorporate fuzzy sets into Artificial Neural Networks\textsuperscript{7}.

This paper discusses a visual inspection methodology, by a neuro-fuzzy approach applied on plates of wood used in pencil industry. In this productive process, the main feature takes into account the visual homogeneity of each plate. Visual homogeneity is the wooden fiber distribution, or knots on the board surface, and it results directly the board quality\textsuperscript{8,9}. The visual homogeneity is used in this paper to define fuzzy variables in automated visual inspection system. The automatic classification time is considered because the system will be applied on the productive process.

2 - WOOD CLASSIFICATION METHODS
Several papers have been published with image processing classical algorithms for defect classification or for node study. The majority of these defects are cracks and holes that, together with nodes, decrease the productivity. Steele et al\textsuperscript{10} use the pixel direction slope for classifying plates without defects from bad ones. Their work detects holes, cracks and nodes. We didn’t...
get good results applying Steele solution on our wooden plates. The pixel direction has small differences because our plates are very similar among them.

Statistical classifiers are traditionally used for defect classification. Koivo & Kim\(^8\) have used mean, median, and minimum and maximum counting for classifying two or three wood classes. One class represents the good ones (homogeneous) and other one represents wood with holes or nodes. Bustler et al\(^{11}\) used statistical classifiers too. They subdivide an image into an array of disjoint rectangular tiles. In each one is processed a pixel intensity transformation (RGB) in terms of an intensity channel, defined as \((R+G+B)/3\) and a color channel defined as \((R-B)/2\). For each tile, four features are defined, namely the pixel intensity and color means and variances. The histograms are used to compute an index for each tile that quantifies how similar their features are. Tiles with relatively low index values are marked as defectives. They detect streaks in the range of 79,2% to 95%. The co-occurrence matrix is applied by Conners\(^9\) to classify eight defects most commonly found in lumber.

Zhu et al\(^{12}\) use computer tomograph (CT) imagery to locate and identify certain classes of defects in hardwood logs. An adaptive filter smoothes each bi-dimensional CT image. Then, a multi-threshold and bi-dimensional segmentation scheme are applied to separate potential defect areas from clear wood areas. Morphological operations, such as erosion and dilation, are applied to a segmented image slice to eliminate rugged boundaries and spurious areas. After this, a three-dimensional image is generated by scene analysis. A set of hypothesis test is applied on this image. The system limitations are the segmentation process, many thresholds and processing time.

By preliminary studies, mean and variance didn’t give sufficient information for classifying our boards. In the papers above, the wooden plates aren’t similar among them. They have strong visual difference. Our classifier works on plates without holes or cracks, but only with texture variation.

### 3 - FUZZY LOGIC

Classical and fuzzy proposition basically differs on their output values\(^{13}\). The domain of a conventional pattern recognition system is a set at true (1) or false(0). Otherwise, the fuzzy domain is set at the interval \([0,1]\) where each output is the membership grade of the variable. Let \(X\) be a space of points with a generic element of \(X\) denoted by \(x\). A fuzzy set \(A\) in \(X\) is defined as:

\[
A = \{ (x, u_A(x)) / x \in X \}
\]

where \(u_A(x)\) is called the membership function of \(x\) in \(A\). The membership function maps each element of \(X\) with a continuous membership value between 0 and 1.

#### 3.1 - FUZZY NEURONS

Kwan and Cai\(^{14}\) define fuzzy neuron having \(N\) weighted inputs, \(x_i\) for \(i = 1\) to \(N\), with \(w_i\) (\(i = 1\) to \(N\)) as the weights, and \(M\) outputs, \(y_j\) for \(j = 1\) to \(M\). All the inputs and weights are real values and the output has real values in the interval \([0,1]\). Each output could be associated with the membership value of a fuzzy concept, i.e., it expresses the pertinence degree to a fuzzy set, of the pattern with the inputs \(\{x_1, x_2, ..., x_N\}\). Thus, we have the figure 1

\[
\begin{align*}
  z &= h[w_1x_1, w_2x_2, ..., w_Nx_N] \\
  s &= f[z - T] \\
  y_j &= g[s] \text{ for } j = 1 \text{ to } M
\end{align*}
\]

**Figure 1** - Fuzzy neuron.

where, \(h[\cdot]\) is the aggregation function (MAX or MIN), \(f[\cdot]\) is the activation function, \(T\) is the activating threshold and \(g[\cdot]\) is the output function (competitive neuron).

The Maximum Fuzzy Neuron does the union of two fuzzy sets \((A \text{ and } B)\)\(^{15}\). It’s called AND neuron. It takes its input in \(X = [x_1, x_2, ..., x_n]\) doing operations over their connections with the weights \(w = [w_1, w_2, ..., w_n] \in [0,1]\) and then does
the global AND with these results. This union operator is defined as:

\[ y = \text{AND}(x; w) \] or
\[ y = T^{n}_{i=1} [x_i \land w_i] \]

where \( t \) and \( s \) are used for representing the AND and OR operation, respectively.

The Minimum Fuzzy Neuron does the intersection of two fuzzy sets (A and B). It’s called OR neuron. Its function is similar to the AND neuron. It takes its input in \( X = [x_1 \ x_2 \ \ldots \ x_m] \) doing operations over their connections with the weights \( w = [w_1 \ w_2 \ \ldots \ w_n] \in [0,1] \) and then does the global OR with these results. This intersection operator is defined as:

\[ y = \text{OR}(x; w) \] or
\[ y = S^{n}_{i=1} [x_i \lor w_i] \]

The task of these neurons is to select among several output levels, one that corresponds to a given input.

The Competitive Neuron compares its state with a threshold (T) gotten from previous layer and it has a binary output (0 or 1). This operator is defined as:

\[ y_m = g[s_m - T] = \begin{cases} 0 & \text{if } s < T \\ 1 & \text{if } s \geq T \end{cases} \]

\[ T = \max \{ [c_1, c_2, \ldots, c_k] \} \]

where the \( t \) is the threshold function \( s_m = \sum w_i \cdot x_i \)

### 3.2 - Fuzzy Rules

The fuzzy rules could be classified in three types, according to its consequent form:

**Type 1:** Fuzzy rules with a constant consequence.

\[ R_i: \text{If } X_i \text{ is } A_{i1} \text{ and } \ldots \text{ and } X_m \text{ is } A_{im} \]

then \( Y \) is \( c_i \)

**Type 2:** Fuzzy rules with a consequent linear combination.

\[ R_j: \text{If } X_j \text{ is } A_{j1} \text{ and } \ldots \text{ and } X_m \text{ is } A_{jm} \]

then \( Y \) is \( g_j(x_1, \ldots, x_m) = b_0 + b_1x_1 + \ldots + b_mx_m \)

**Type 3:** Fuzzy rules with a consequent fuzzy set.

\[ R_k: \text{If } X_k \text{ is } A_{k1} \text{ and } \ldots \text{ and } X_m \text{ is } A_{km} \]

then \( Y \) is \( B_k \)

where \( X \) and \( Y \) are respectively the input and output variables. The linguistics terms (\( A_{im} \)) are fuzzy sets with a specific function (triangular, sigmoidal, trapezoidal). The \( c_i \) term is a constant value. The \( g_j \) term is a linear array of input variables, and the \( b_m \) terms are the constant coefficients. The \( B_k \) term shows another fuzzy set.

### 4 - Fuzzy Neural Network

An Artificial Neural Network (ANN) is a set of nodes connected by direct links. Each node performs a particular function on incoming signals using a set of parameters specific to this node. The ANN’s are classified, by their connection types, in feed-forward and recurrent networks. In the feed-forward type, the signal flows from one output to one input in the next layer. Networks that are not strictly feed-forward, but include direct or indirect loops of connections, are often called recurrent networks. Among the several network architectures (Hopfield, Boltzmann’s Machine, Kohonen), the backpropagation neural network is the most used, because its learning usually converges to a desired output.

Fuzzy-neural networks can be divided into two main categories. One group of neural networks for fuzzy reasoning uses fuzzy weight in the neural network. In another group, the input data are fuzzified in the first or second layer, but the neural network weights are not fuzzy. Fuzzy min-max neural networks are built using the following layers:

- **Input Layer (Fuzzification):** this layer transforms the crisp signal into a membership value through a membership function.
- **Rule Layer MIN (MAX):** the nodes in this layer perform fuzzy AND/OR. The AND/OR rules can be on distinct layers or on the same layer. The AND/OR operations can be in the same rule, too.
- **Output Layer (Defuzzification):** this layer transforms the fuzzy value into a crisp one. Kwan and Cai use in this layer competitive fuzzy neurons in which only one node shot.

Some researchers use another layer as, for example, the matching layer. This layer has one input from some input linguistic node, and its output feeds the rule nodes. The output in this layer indicates the matching degree of input and fuzzy weight.

### 5 - Wooden Plate Classification and Fuzzy Variables
According to the industrial production and by taking into account the product quality and the market demand, the wooden plates used in this paper can be classified, concerning their visual homogeneity, as:

- Plate A: These plates are the best ones for pencil manufacturing. They have a better visual homogeneity. They don’t have nodes, stripes and a dark area on their surface (figure 2).
- Plate C: They are intermediate plates, and they have longitudinal stripes. These dark stripes represent a low percentage regarding the total board area (figure 3).
- Plate S: They are the worst plates for pencil manufacturing. They have areas with different pigments. Sometimes it is difficult to visually distinguish which area prevails. They will be reject in an industrial production line (figure 4).

The ideal histogram of a wooden plate will be a pulse in a certain level. This implicates that the plate possesses only a color in all its surface. By sampling the A, C and S plates, we could see by histogram analysis, that “S” class has the “range width” wider than “A” and “C” classes. So, this class has less pixel quantity by each level. “A” class has the “range width” slightly narrower than “C” class.

The most of authors have used statistical methods for classifying textures. One example is the use of co-occurrence matrices.23,24 The inconvenience with these methods is the great amount of processing time during the classification. The proposed method described in this paper is bounded by real time processing. We have to classify at least 150 plates by minute, due to industry needs.
We extracted the “range width” from histogram, taking into account the gray levels greater than a “K” value, as showed in Fig. 5. This feature gives the plate homogeneity. A narrow “range width” shows that the plate has a small quantity of gray levels. The more the “range width” increase, the more gray levels will be present on the plate surface. So, this quantity is used as a fuzzy variable giving the plate membership related to visual homogeneity. The second fuzzy variable is extracted from histogram taking the point quantity at “elbow region”. This region is gotten checking the peak level from histogram and thus, searching for the beginning of region slope. This quantity shows darker levels than the gray levels under the range width. This means nodes, stripes or defects on the plate surface. The fuzzy variable based on the “elbow region” is an inhibiting variable because this feature is stronger for “S” class. The third fuzzy variable is needed for discriminating the “C” from “A” class. The visual difference between them is the stripes in “C” plate. We do four transversal scans and then we count the number of crossed stripes. For the majority of the “A” class this number is small and for the “C” class this number is big. Moreover, the number of stripes in the “C” class is the same in any scan.

5.1 FUZZIFICATION

The neural network input layer has four nodes. The first one is the range width, the second one is the elbow region, the third one is related with the global contrast and the last one is the number of detected stripes. By the histogram of 100 plates, front and back, we get the range width distribution showed in Fig. 6. By this graphic, we get the membership functions for the fuzzy variable named range width, as showed in Fig.7; a bell shaped function and a sigmoidal function.

The second input is the pixel number of elbow region. That’s a good separator of bad plates (“S” class). It’s an inhibitor input for the fuzzy rules. It does the general rules dependent from a specific input. The third input is gotten by contrast enhancement and then by counting pixels in dark region. This pixel quantity gives the membership functions showed in Fig. 8.
The fourth input is gotten by taking the mean of the four quantities generated by the four transversal scans. That’s a good variable for discriminating between “A” class and “C” class plates. The membership function for this fuzzy variable is showed in Fig. 9. The inputs are then applied to the Neural Network of Fig. 10.

Figure 8 - Membership functions for global contrast.

Figure 9 - Membership functions for stripes quantity.

Figure 10 - Neuro-fuzzy network used.
5.2 - FUZZY IMPLEMENTATION

The second network layer (fuzzyfication) processes the input through the membership function from each class (A, C, and S). For fuzzy algorithms implementing we need a set of operators that will handle the fuzzy quantities described by the membership functions. These operators are the intersection (AND), the union (OR) and the implication (IF ... THEN...). The fuzzy rule layer of the network uses the union property described by Jang and Sun, that is:

\[ u_c(x) = \max (u_A(x), u_B(x)) \] where A, B, and C are fuzzy sets.

The minimum fuzzy neurons are used in the output layer of membership functions, that is:

\[ u_c(x) = \min (u_A(x), u_B(x)) \] where A, B, and C are fuzzy sets.

The last network layer classifies the plates by only one neuron. We are using the sigmoidal function setting up three bands. These bands refer to the result of classification for each plate and have the following values, as we can see from Fig. 11:

- from 0.0 to 0.399 - “S” class
- from 0.4 to 0.699 - “C” class
- from 0.7 to 1.0 - “A” class

![Figure 11 - Neural network output.](image)

5 - EXPERIMENTAL RESULTS

Training neuro-fuzzy classifier consists of selecting a training set \( T \) that has 20 ordered pairs:

\[ \{P^k, c_i\} \quad k = 1, 2, \ldots, 20 \]

\[ i = A, C \text{ and } S \]

where \( i \) are classes, \( P^k \) is an input pattern and \( c_i \) is its class. Input patterns have four dimensions and they can be described by:

\[ P^k = \{p^k_1, p^k_2, p^k_3, p^k_4\} \]

Training proceeds by successively presenting ordered pairs from \( T \) to the neuro-fuzzy system. The system learns by a backpropagation algorithm. The desired output was taken from the intermediate point of output range for each plate (see figure 11) with an error of 0.01%.

With a test set of 89 plates the system processes 178 images, considering front and back plate sides, in real time. The Fig. 12, 13 and 14 show the network output for classifying plates from “A”, “C”, and “S” classes, respectively.
By classifying “A” class the result can be showed at figure 12. From figure 12, we find 45 plates classified as “A” class, 6 plates as “C” class and only one as “S” class.

**Figure 12** - Plate “A” classification.

The results showed in figure 13 were obtained for “C” class. We got 42 plates classified as “C” class, 18 plates as “S” class and 2 plates as “C” class.

**Figure 13** - Plate “C” classification.

The analysis of plates “S” shows the results presented in figure 14. We obtained 56 plates classified as “S” class and only 2 as “C” class.

**Figure 14** - Plate “S” classification.

In order to check the network accuracy and reliability, we have taken one plate from each class, randomly, and submitted it 100 times to the network input. The Fig. 15, 16, and 17 show the results to “A”, “C”, and “S” class respectively. It can be seen the high accuracy (repeatability) to “S” class. This class is easy to classify by visual inspection, due to node distribution on the surface. The same result was gotten in our implementation.

The variation of the results inside figures 15 and 16 is irrelevant. This variation occurs because of light variation. The “A” and “C” classes got good results, classifying the same plate within the same class all the time, without any error. This is an excellent result because these plates are difficult to classify by human visual inspection.
We create a simple and effective neuro-fuzzy system to classify patterns. The system quickly and easily classifies wooden boards into three classes. With the 178 images gotten from the 89 plates we did the classification task. The errors and right classification percentage are showed in Table 1.

<table>
<thead>
<tr>
<th>Class</th>
<th>Right Classification</th>
<th>Wrong results</th>
<th>% right</th>
<th>% errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>45</td>
<td>7</td>
<td>86.5</td>
<td>13.5</td>
</tr>
<tr>
<td>C</td>
<td>42</td>
<td>26</td>
<td>61.7</td>
<td>38.2</td>
</tr>
<tr>
<td>S</td>
<td>56</td>
<td>2</td>
<td>96.5</td>
<td>3.5</td>
</tr>
<tr>
<td>Total</td>
<td>143</td>
<td>35</td>
<td>80.3</td>
<td>19.7</td>
</tr>
</tbody>
</table>

These results show the excellent classification rate for “S” class plate, and a good classification rate for “A” class plate. The “C” class plate could be classified as “S” class or “A” class depending on demand. If demand is high, high values from “C” class could classify the plate as “A” class and low values as “S” class. If demand is low, all “C” class could be classified as “S” class. A new input could be added to the neural network named demand. The human operator could introduce this value controlling the network output. We could control illumination over the scene and increase network reliability.

The MIN-MAX neurons are simple mathematical operations. This assures a reasonable speed during process production. The processing time for one plate classification was about 0.39 seconds. Our system is based on a 486 processor with a Data Translation frame grabber, and Hitachi camera mounted over a conveyor belt. It classifies 153 plates by minute. This processing time could be reduced by changing the processor. Nevertheless, our system, with the implemented features, shows compatibility with industry needs.

7 - REFERENCES


