Cork Quality Classification System using a Unified Image Processing and Fuzzy-Neural Network Methodology

Joongho Chang, Gunhee Han, José M. Valverde, Norman C. Griswold, Senior Member, IEEE, J. Francisco Duque-Carrillo, Member, IEEE, and Edgar Sánchez-Sinencio, Fellow, IEEE

Abstract—Cork is a natural material produced in the Mediterranean countries. Cork stoppers are used to seal wine bottles. Cork stopper quality classification is a practical pattern classification problem. The cork stoppers are grouped into eight classes according to the degree of defects on the cork surface. These defects appear in the form of random-shaped holes, cracks, and others. As a result, the classification cork stopper is not a simple object recognition problem. This is because the pattern features are not specifically defined to a particular shape or size. Thus, a complex classification form is involved. Furthermore, there is a need to build a standard quality control system in order to reduce the classification problems in the cork stopper industry.

The solution requires factory automation meeting low time and reduced cost requirements. This paper describes a cork stopper quality classification system using morphological filtering and contour extraction and following (CEF) as the feature extraction method, and a fuzzy-neural network as a classifier. This approach will be used on a daily basis. A new adaptive image thresholding method, and a fuzzy-neural network as a classifier. This approach will be used on a daily basis. A new adaptive image thresholding method using iterative and localized scheme is also proposed. A fully functioning prototype of the system has been built and successfully tested. The test results showed a 6.7% rejection ratio. It is compared with the 40% counterpart provided by traditional systems.

The human experts in the cork stopper industry rated this proposed classification approach as excellent.

Index Terms—Cork, feature extraction, fuzzy neural networks, image binarization, image enhancement, image processing, image shape analysis, image size analysis, morphological filters, neural networks.

I. INTRODUCTION

Cork is the biological name of a tissue produced by most of trees around the trunk. The cork oak is exceptional in that the cork layer achieves enough thickness to be processed with industrial purpose. Indeed, the most important and known cork derivative product is the cork stopper, which is almost exclusively used to seal wine. This is why, unlike any other natural or man-made material, cork provides simultaneously three properties to be used as a unique material for sealing wine [1]. First of all, a cork stopper avoids leakage even in the presence of irregularities in the bottle neck. Second, it presents an excellent chemical inertia, which avoids any reaction with the wine which changes wine taste. And finally, a more subtle characteristic, but indeed not less important, consists of its ability to allow gaseous exchange with the outside atmosphere, which permits the wine quality to improve as time goes by. This process is known as maturation.

However, as a natural material, cork is heavily heterogeneous and therefore, there are no two cork stoppers alike. This has been a severe handicap as far as quality control is concerned. The quality of a cork stopper is decided by the complex combination of defect features described above.

The proposed system was designed to be used by both the wine producers as well as the wine manufacturers. It is compared with the 40% counterpart provided by traditional systems. The human experts in the cork stopper industry rated this proposed classification approach as excellent.

The cork images, respectively. There exist a certain amount of classification. Most insects defects are caused by ants penetrating from top to bottom of the cork stopper and they appear as small round hole which is exceptionally dark. In this case, a cork stopper should be considered as a worst class because it may cause leakage of wine through the defect.

Besides above major defects, there are minor defects such as canals which are counted as the second defect features. In most cases, eight different classes of cork stoppers are considered in the cork industry. Corks are classified based on the complex combination of defect features described above.

Fig. 1 shows some example images of cork stoppers. Fig. 1(a) is the first class image which looks clean and we can hardly find any defects. Fig. 1(b) and (c) shows third and fifth class cork images, respectively. There exist a certain amount of holes, cracks, and canals.
the surface. Fig. 1(e) has long and sharp crack and Fig. 1(f) has insects on the top and bottom of stopper while the cylinder surface is clean as much as first class case.

As a consequence of the high heterogeneity, traditionally the quality control of cork stoppers has been carried out by human experts. However, the lack of universally accepted classification criteria causes important commercial discrepancies between cork manufacturers and wine companies. Recently, electronic cork classification systems have been introduced in the market. In such systems the cork image is first captured by one or several charge-coupled device (CCD) cameras. After the image thresholding, only three features are identified for each sample even on the most sophisticated systems. These three features are the total number of holes, the size of the biggest hole, and the maximum crack length. The system makes the decision according to the worst evaluated feature among these three features. The system performance is acceptable for high quality stoppers. However, for intermediate and low quality stoppers, the number of misclassified samples is large (around a 40% in misclassification ratio). Due to the poor performance of this system, the stoppers should be reevaluated by human experts.

In this paper, we present a cork stopper quality classification system based on more advanced feature extraction methods and a fuzzy-neural network. The proposed system achieves significantly higher performance when it is compared to the traditional classification systems. The proposed system is to be adapted in actual cork factories for the everyday usage. This paper is organized as follows: A general description of the classification system is provided in Section II. The feature extraction methodologies are described in Section III. Section IV deals with the fuzzy-neural network classifier. Finally, our experimental results and conclusion are given in Sections V and VI, respectively.

II. SYSTEM OVERVIEW

In our system, a cork stopper image, which has a size of 288 \( \times \) 263, consists of two parts. One corresponds to the stopper cylinder surface which has a size of 288 \( \times \) 170, while the other is the top and bottom surface which has a size of 288 \( \times \) 93. Fig. 2(a) shows these parts in a stopper and Fig. 2(b) illustrates the image capturing system. The cylinder surface image is obtained by a linear CCD camera while a stopper is rotated by a step motor which is controlled by a PC. The top and bottom surface images are obtained using two other CCD cameras. By using this image capturing system, the gray images shown in Fig. 1 are generated.

An adaptive image thresholding is introduced to overcome the irregular light condition and nonuniform cork albedo status in the cylinder surface images. The morphological filtering [2], [3] is used as a basic feature extraction method. This method extracts the information about size, distribution and the area of defects. The contour extraction and following (CEF) [2] is used to detect thin and long crack which can not be detected by morphological filtering. Several directional template filters are used prior to CEF to improve the performance of CEF.

The detailed functionalities of the system is illustrated in Fig. 3. Input features #1–#5 are the normalized values of the summation of all pixel points in each 0 \( \times \) 0, 3 \( \times \) 3, 5 \( \times \) 5, 7 \( \times \) 7, 11 \( \times \) 11 morphological erosion filtered images for a cylinder surface. A 0 \( \times \) 0 eroded image is the original binary image which no erosion occurs. Input feature #6 is the normalized length of the longest defect on the cylinder.
surface obtained by CEF. Input feature #7 is the normalized length of the longest defect obtained by CEF, which locates very close to the top and bottom edge of the cylinder. Input feature #8 is the normalized value of the summation of all pixel points after the $3 \times 3$ morphological erosion filtering for the top and the bottom. For the top and bottom, a simple image thresholding (a single threshold gray level around 90) without the adaptive thresholding is used. Insects defects is obtained only from the top and bottom surface image. To extract only insects, low threshold gray level (around 45) is used in image thresholding without adaptive method because the insects defects are usually much darker than other defects. After image thresholding for insects defects, we use CEF to detect any defects which is large enough to be considered as insects defects. The fuzzy multilayer perceptron (MLP) neural network [4] is trained using these eight input features for each stopper image. The insects defects are not used as the input feature for neural network because, regardless of the neural network output result, if we find any insects defects on the image, then the stopper is to be classified as the worst class.

III. FEATURE EXTRACTION

A. Adaptive Thresholding

Defects from the cork background are simply extracted by means of the value of gray level. In most cases, defects have from 50 to 90 gray-level intensity while the background counterpart has from 100 to 130 gray-level value. The purpose of gray level thresholding is to extract pixels which represent objects from the background of the image. Unfortunately, it is very difficult in general to find a single threshold which is best for an arbitrary gray-level image [5], [11]. There have been many approaches to find an optimal threshold level for certain image cases. Iterative selection procedures [6] are methods where a guess is made at the object and background levels and then, thresholds are selected iteratively while adjusting the guesses at each iteration. The two-dimensional correlation methods [7] are methods where all reasonable thresholds are applied, and the thresholded images in each case are correlated with the original image to find an optimal threshold.

However, in the case that the illumination over the image is different, owing to the nonuniformly distributed light condition, the different background color, albedos or nonuniform surface curvature, which are very frequent circumstances in industrial plants dealing with cork stoppers, the result can be that the objects in certain parts of the image may be lighter or darker than any other part. Therefore, it is inadequate to apply a single threshold to the whole image. If a single threshold is not acceptable, then the remaining option consists of using a number of different localized thresholds over small subregions of the image. This method decides threshold values based on the local properties of the image. For the localized thresholding method, several approaches have been proposed [8]–[10]. The intensity gradient based thresholding methods [8] use the gradient with surrounding pixels as a measure of the decision for the local threshold. Other approaches appeal to local histogram information. These methods merge the local histogram information by statistical method or connectionist network method [9], [10]. Each approach has its own advantages or disadvantages.

In our case, the cork stopper image can be said to be a typical poorly illuminated image. As shown in Fig. 4, the histogram of the original image is too dense and complicated to decide the optimal threshold to distinguish defects from the background. However, if we zoom in into each local area with different widow sizes, then we observe the different shapes of histograms for each local area. By the transformation given in (1), we can generate more distance between the gray levels of the defect points and the background, which usually has a higher gray level average for each local area. That is, the image is enhanced by splitting the gray level of each pixel from its local neighborhood average using iterative method. In order to preserve the broad neighborhood information and narrow neighborhood information efficiently, the window size decreases as iteration increases as shown in Fig. 5.

The main algorithm is given in Fig. 6. For each local image divided by the window size $W_k$ at iteration $k$, the average value of gray level for every pixel points in the windowed
area, \( \text{ave}(x, y, W_z) \) is calculated. Around this average, we simply take a linear transformation as a function of the distance between the average and the gray level of each pixel within the windowed area. If the distance between the average and the gray level of a pixel is large, then the gray level of that pixel point moves further away from the average to the opposite direction. On the contrary, if the distance is small, then the gray level of the pixel point would move slightly.
The transformation can be expressed as follows:

\[
A_k(x, y) = A_{k-1}(x, y) + \mu \{ \text{ave}(x, y, W_k) - A_{k-1}(x, y) \}
\]

\[
\text{ave}(x, y, W_k) = \frac{1}{W_k^2} \sum_{i \in N_x(W_k)} \sum_{j \in N_y(W_k)} A_{k-1}(i, j)
\]

\[
N_x = \left\{ x = \left[ \frac{W_k}{2} \right], \cdots, x + \left[ \frac{W_k}{2} \right] \right\}
\]

\[
N_y = \left\{ y = \left[ \frac{W_k}{2} \right], \cdots, y + \left[ \frac{W_k}{2} \right] \right\}
\]

\[
0 < \mu < 1, \quad k = 1, 2, \cdots, N \quad (N = \text{Total # of iterations})
\]

(1)

where \(A_k(x, y)\) is the gray level of a pixel \((x, y)\) at \(k\)th iteration and \(\mu\) is the update rate. \(N_x\) and \(N_y\) is the index set for window area. This routine is iterated several times with different sizes of window. After some iteration, the final well separated histogram between defect and background gray level is obtained as shown in Fig. 4. We can obtain different degree and style of image enhancement by adjusting the several factors such as update rate \(\mu\), total number of iteration, initial widow size, and the decreasing rate of the window size \(\alpha (0 < \alpha < 10)\). In our case, we set \(\mu\) as 0.4, total number of iterations as ten, initial window size as 100, and \(\alpha\) as eight. After obtaining the enhanced image, an appropriate global threshold gray level, which is indicated as \(\delta\) in Fig. 6, for the entire image is easily decided. We used a gray level around 20 as a global threshold since most of the gray level of defect pixel points are moved very close to zero.

By this new proposed thresholding method, we obtain well locally defined binary images for the complicated cork images despite the irregularity of image. An example of the enhanced image and thresholded image are shown in Fig. 7. Fig. 7(a) is a typical image whose upper part is darker than the lower part. Fig. 7(c) and (d) shows thresholded images by the proposed adaptive thresholding method and by a conventional single
thresholding method, respectively. In Fig. 7(c), the actual defects were extracted more efficiently than the conventional method.

B. Morphological Filtering

Morphological operations are derived from the branch of mathematical analysis called Minkowski algebra [2]. Mathematical morphology is an approach of image processing based on set-theoretic concepts of shape. Morphological image processing is useful for shape analysis, feature extraction, and nonlinear filtering [3]. Basically, morphological image processing is the analysis of the geometrical relationship between the image and a smaller image called a structuring element. Among the merits of morphological image processing, it can be pointed out that it allows important geometrical image features to be preserved with simple algorithms which may lead to parallel implementation structures. The disadvantages can be said that the processing is nonlinear and there is a lack of analytical criteria for choosing the structuring elements. The set-theoretic field of study is based on two simple but powerful binary operations called dilation and erosion. As the basic feature extraction method in our case, erosion operation is used.

If \( f(x,y) \) is a digitized binary image and \( S(m,n) \) a binary structuring element then erosion of \( f \) by \( S \), denoted by \( E(f,S) \), is defined by

\[
e(x,y) = E(f,S) = \bigwedge_{(m,n) \in \text{Domain}(S)} \text{Trans}(f; m,n)
\]  

where \((m,n)\) are elements of the domain of \( S \) and \( \text{Trans} \) operation denotes the translation of image \( f \) with the index of \( m,n \).

In the erosion, the digitized binary image \( f \) is translated by an activated pixel of \( S \) to produce a new image \( e(x,y) \). This processing is repeated for each activated pixel of \( S \) and then the intersection is taken. In a binary image, erosion has the effect of shrinking the image.

If we erode the image with a 3 \( \times \) 3 rectangular structuring element, then the defects smaller than this rectangular window would be removed. Similarly, if we erode with a 5 \( \times \) 5 rectangular structuring element, then the defects smaller than this rectangular window would be removed. By repeating this process with different size of structuring elements, \( 3 \times 3, 5 \times 5, 7 \times 7, 11 \times 11 \), the proper information about the amount and size of defects on an image is obtained. If there exist some big defects on the image, then all the filtering results from 0 \( \times \) 0, 3 \( \times \) 3, 5 \( \times \) 5, 7 \( \times \) 7, 11 \( \times \) 11 would have large values. If only a lot of small defects exist on the image, then, even though the filtering results from 0 \( \times \) 0, 3 \( \times \) 3 would have large values, the filtering results would decrease rapidly as the size of structuring element increases. From the tendency of values for each filtering with different structuring element size, we can obtain the size information of defects on the image. As the input features for fuzzy neural network classifier, the filtering result \( F_k \), which is given by the following expression, is used:

\[
F_k = \sum_{i=0}^{M} \sum_{j=0}^{N} I_k(i,j) \quad k = 0, 3, 5, 7, 11
\]

where \( F_k \) is the summation of every pixel points in each eroded image \( I_k \) with different structuring elements \( k \times k \) on the \( M \times N \) image. The examples of eroded images and corresponding \( F_k \) values for each \( k \) are shown in Fig. 8.

C. Directional Template Filtering

When the CEF processing is performed, if there are several small defects around the main thin and sharp defects, they would be merged together and it becomes difficult to detect the very thin and sharp crack defects correctly. For such purpose, several templates of different shapes, which are shown in Fig. 9, are used in order to filter out some trivial small defects. These filters were designed to detect only thin and long connected objects in any directional components. After this directional template filtering, most of trivial defects are removed and simultaneously, thin and long components are emphasized as illustrated in Fig. 10.
Some specific shapes of defects, such as very thin and sharp crack defects, are very difficult to be detected by simple morphological erosion filtering. Because in many cases these defects have thickness of less than two or three pixels, and even when we apply a small $2 \times 2$ or $3 \times 3$ structuring element, it may lose the connectivity, which is very important in the sharp crack defect case. Therefore, we incorporated CEF [2] as a robust additional methods to detect these defects for such a purpose. Contours have linked edges that characterize the shape of an object. They are very useful in computation of geometrical features such as size or shape. Conceptually, a contour can be found by tracing the connected edges. On a rectangular grid, a pixel is said to be four or eight connected when it has the same properties as one of its nearest four or eight neighbors.

In our case four connectivity characteristic is used. By using a small template shown in Fig. 11(a), only a contour of each object is obtained. After obtaining one-pixel-thick contours of objects, the contour following is performed to measure the perimeter of closed contours. The algorithm that carries out this task is given as follows [refer to Fig. 11(b)].

1) Start at any point A on the boundary.
2) Find the nearest boundary pixel and move there.
3) Count the number of pixels.
4) Continue until the point goes back to starting point A.

If there exists any large perimeter of object in an image then we may conclude that there is big hole or long and sharp crack defects on the cork surface. Among the several perimeters of objects, the maximum perimeter is passed to the classifier only when it exceeds a certain threshold which is decided by a user to adjust the behavior of the total system. In Fig. 12, the result of the contour extraction for a cork image is illustrated.

IV. CLASSIFIER

The decision making of the cork quality by human experts is based on very complicated rules such as “if the total area of hole is $A$, then it gives a penalty $f$ to the decision, if the total area of crack is $B$, then it gives a penalty $g$ to the decision, if a cork stopper has $f, g, \ldots, k$ penalties then the final decision of the cork is $p$th class.” To set all these complicated rules, many efforts and time consuming discussions would be required by human experts. In practice, carrying out this task would be even harder if we take into account the different criteria existing among the experts about the cork quality. For such reasons, there has been a lack of normalization in cork stopper quality control. As a consequence, conflicts between cork manufacturers and wine companies have occurred frequently.
For such a complicated decision rule problem, the solution is based on the use of a neural-network paradigm which can mimic the human reasoning [12]. The benefits of the neural network is the generalization ability [13] about the untrained samples due to the massively parallel interconnections and easiness of implementation simply by training with training samples for any complicated rule or mapping problem. However, the performance of neural network is trustworthy upon the assumption that we have enough number of training samples [14].

It may be mentioned that human reasoning is somewhat fuzzy in nature, especially in the case of cork quality classification. This classification problem is different from other classification problems such as character recognition. In the character recognition, the desired class to which a pattern belongs is clear. However, in the cork classification case, the decision can vary from person to person or time to time. Even the same person and at the same time, he or she can not make the clear decision that a cork stopper absolutely belongs to a certain class. The utility of fuzzy sets [15]–[17] lies in their ability to model the uncertain or ambiguous data so often encountered in real life. Therefore, to enable a system to take care of real-life situations in a manner more like humans, the concept of fuzzy sets into the neural network has been incorporated. It needs to be noted that although fuzzy logic is suitable for propagating uncertainty, it may cause an increase in the amount of computational burden when compared with conventional digital logic system. In the case of cork quality classification system, if five different structuring elements in conventional digital logic system. In the case of cork quality classification system, if five different structuring elements in the morphological filtering feature extraction are used and with three membership functions for each feature, then we should build totally $5^3 = 125$ rules for a decision. But it is very difficult to build 125 rules observing the sample feature space and especially, it needs so much feedback to adjust the rules intuitively. This is known as the “tuning” process in fuzzy methods. Also, in the recall phase after tuning, we should pass through all the 125 rules composed with many Min or Max operations. It therefore gives the system a heavy burden of computation. These problems can be suitably reduced by using fuzzy neural network which has training capability.

An MLP classifier, with a backpropagation training algorithm which incorporates concepts from fuzzy sets at the training stage, is used. The main idea implemented in our system originated from Pal [4]. However, in their approach, they converted the input features to the fuzzy components in the input vector, which therefore consists of the membership values on overlapping partitions of linguistic properties such as low, medium, and high corresponding for each input feature. This causes the expansion of the total number of connections between the input layer and the hidden layer. We can develop better generalization ability than the conventional MLP network with the manipulation only on the output layer with the fuzzy desired output, which is much simpler than the approach in [4].

In general, the MLP passes through two phases, training and testing. During the training phase, supervised learning is used to assign the output membership values ranging in [0, 1] to the training input vectors. Each output from MLP may be assigned with a nonzero membership instead of choosing the single node with the highest activation. It allows modeling of fuzzy data when the feature space involves overlapping pattern classes, such that a pattern point may belong to more than one class with a nonzero membership. During training, each error in membership assignment is fed back and the connection weights of the network are appropriately updated. The backpropagated error is computed with respect to each desired output, which is a membership value denoting the degree of belongingness of the input vector to a certain class. The testing phase in fuzzy MLP is equivalent to the conventional MLP.

In the case of $n_k$-class problem with $n$-dimensional feature space, let the $n$-dimensional vectors $\mathbf{O}_{kj}$ and $\mathbf{V}_{kj}$ denote the mean and the standard deviation for the $j$th input feature respectively of the numerical training data for the $k$th class. The weighted distance, $Z_{ik}$, of the training pattern vector $\mathbf{F}_i$ from the $k$th class is defined as

$$z_{ik} = \sum_{j=1}^{n} \left( \frac{F_{ij} - O_{kj}}{V_{kj}} \right)^2 \quad \text{for} \quad k = 1, \ldots, m \quad \text{and} \quad j = 1, \ldots, n$$

(4)

where $F_{ij}$ is the value of the $j$th input feature component of the $i$th pattern point. The weight $1/V_{kj}$ is used to take care of the variance of the classes so that a feature with higher variance has less significance in characterizing a class. The membership of the $i$th pattern to class $C_k$ is defined as follows:

$$\mu_k(\mathbf{F}_i) = \frac{Z_{ik} - \min_k(Z_{ik})}{\max_k(Z_{ik}) - \min_k(Z_{ik})} \quad \text{for} \quad k = 1, \ldots, m.$$ 

(5)

Obviously $\mu_k(\mathbf{F}_i)$ lies in the interval [0,1]. Here, the larger the distance of a pattern from a class, the lower its membership value to that class. Except for the fuzzy membership desired values in the output layer, the training method and network structure is equivalent to the conventional MLP classifier. The training scheme for fuzzy MLP neural network is shown in Fig. 13.

![Fig. 13. The training for the fuzzy MLP neural network.](image)
V. EXPERIMENTAL RESULTS

The software was implemented on a SUN sparc workstation after storing all the sample images obtained by an image capturing system on the PC. Seventy samples for each of the eight classes are used for training and 80 samples for each class are used for testing. Typical examples of eight-dimensional features and insects defect are given in Table I. It includes the example input features for every eight class cases and also includes the eighth class—holes, cracks, and insects shown in Fig. 1(d)–(f). All the values in the table are the normalized value within [0.0, 2.0]. We may evaluate the performance of cork quality classification system by two methods. The first method is the confusion matrix indicating whether the classification tendency is reasonable or not. Due to the unique characteristic of cork stopper classification (i.e., There is no absolute desired class), it is not proper to count the percentage of correct classification for performance evaluation. On the confusion matrix, if we have a normal distributed matrix with few outliers, centered on the diagonal direction points, the classification can be said to be reasonable. The confusion matrix from 80 test samples for each class and its graphical representation are given in Table II and Fig. 14, respectively. Diagonal dominance is clearly observed from Fig. 14. The second evaluation method is the percentage of rejection by the human experts. The human experts reevaluated the result from our proposed system to obtain the total percentage of rejections. In the experiment with 80 testing samples for each class, we had around 6.7% of rejection ratio which is acceptable in a real industry environment.

VI. CONCLUSION

The development of a cork quality classification was a system integration problem in order to build a system solution intended to perform a specific task. By using this automatic classification system, we may reduce cost, time and also many.

### TABLE I
EXAMPLES OF EIGHT-DIMENSIONAL INPUT FEATURES

<table>
<thead>
<tr>
<th>Feature #1</th>
<th>Feature #2</th>
<th>Feature #3</th>
<th>Feature #4</th>
<th>Feature #5</th>
<th>Feature #6</th>
<th>Feature #7</th>
<th>Feature #8</th>
<th>Insect defect</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st class</td>
<td>0.3458</td>
<td>0.4521</td>
<td>0.1209</td>
<td>0.0987</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0310</td>
</tr>
<tr>
<td>2nd class</td>
<td>0.4492</td>
<td>0.5321</td>
<td>0.3542</td>
<td>0.2189</td>
<td>0.2342</td>
<td>0.0</td>
<td>0.0</td>
<td>0.2143</td>
</tr>
<tr>
<td>3rd class</td>
<td>0.7600</td>
<td>1.0229</td>
<td>0.6321</td>
<td>0.3098</td>
<td>0.2673</td>
<td>0.0</td>
<td>0.0</td>
<td>0.2721</td>
</tr>
<tr>
<td>4th class</td>
<td>0.9142</td>
<td>1.1892</td>
<td>1.0381</td>
<td>0.8907</td>
<td>0.5428</td>
<td>0.0</td>
<td>0.0</td>
<td>0.2897</td>
</tr>
<tr>
<td>5th class</td>
<td>0.8952</td>
<td>1.1237</td>
<td>0.8552</td>
<td>0.7892</td>
<td>0.4830</td>
<td>0.4498</td>
<td>0.3296</td>
<td>0.3749</td>
</tr>
<tr>
<td>6th class</td>
<td>1.1761</td>
<td>1.1935</td>
<td>1.1132</td>
<td>1.0049</td>
<td>1.2375</td>
<td>0.5482</td>
<td>0.0895</td>
<td>0.2245</td>
</tr>
<tr>
<td>7th class</td>
<td>1.3425</td>
<td>1.2593</td>
<td>1.2786</td>
<td>1.3586</td>
<td>1.4498</td>
<td>1.0872</td>
<td>0.5892</td>
<td>0.8821</td>
</tr>
<tr>
<td>8th class-holes</td>
<td>1.6917</td>
<td>1.5983</td>
<td>1.5793</td>
<td>1.7224</td>
<td>1.9289</td>
<td>1.3125</td>
<td>0.8241</td>
<td>0.7762</td>
</tr>
<tr>
<td>8th class-cracks</td>
<td>0.5711</td>
<td>0.7792</td>
<td>0.3546</td>
<td>0.1729</td>
<td>0.2100</td>
<td>2.0000</td>
<td>2.0000</td>
<td>0.1164</td>
</tr>
<tr>
<td>8th class - insects</td>
<td>0.4628</td>
<td>0.5267</td>
<td>0.4492</td>
<td>0.3008</td>
<td>0.2831</td>
<td>0.0</td>
<td>0.0</td>
<td>0.6599</td>
</tr>
</tbody>
</table>

### TABLE II
CONFUSION MATRIX

<table>
<thead>
<tr>
<th></th>
<th>class 1</th>
<th>class 2</th>
<th>class 3</th>
<th>class 4</th>
<th>class 5</th>
<th>class 6</th>
<th>class 7</th>
<th>class 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>class 1</td>
<td>38</td>
<td>25</td>
<td>8</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>class 2</td>
<td>14</td>
<td>36</td>
<td>17</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>class 3</td>
<td>3</td>
<td>13</td>
<td>38</td>
<td>22</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>class 4</td>
<td>0</td>
<td>3</td>
<td>8</td>
<td>25</td>
<td>27</td>
<td>13</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>class 5</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>14</td>
<td>28</td>
<td>21</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>class 6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>16</td>
<td>38</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>class 7</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>12</td>
<td>16</td>
<td>26</td>
<td>22</td>
</tr>
<tr>
<td>class 8</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>12</td>
<td>21</td>
<td>43</td>
</tr>
</tbody>
</table>

Fig. 14. Graphical presentation for confusion matrix in Table II.
conflicts in the industry since this system may be used as a standard for the cork stopper quality control. We started this project with very simple ideas such as the morphological filtering method and the neural-network classifier. However, as we got more involved in this project, we faced a lot of critical problems that should be solved to obtain a better classification performance, such as the image thresholding and feature extraction for some specific defects like very thin and sharp cracks. These problems were solved by developing a new style of adaptive thresholding method and by incorporating CEF method. With the fuzzy MLP neural network, we could build the classification rules very easily by simply training the network with enough number of sample images and we could implement the nature of fuzzy classification into cork quality.

The evaluation results for the system seemed very promising and satisfied the human experts in the cork stopper industry. The remaining problems to make the system implemented in the real industry is a hardware implementation. The processing time to make a decision for a cork stopper image is very critical. It should be done in real-time-processing. We may use a coprocessor or digital signal processor accelerator board to speed up the processing time in the PC environment. However, our final goal in this project is to implement the nature of fuzzy classification into cork quality.

The evaluation results for the system seemed very promising and satisfied the human experts in the cork stopper industry. The remaining problems to make the system implemented in the real industry is a hardware implementation. The processing time to make a decision for a cork stopper image is very critical. It should be done in real-time-processing. We may use a coprocessor or digital signal processor accelerator board to speed up the processing time in the PC environment. However, our final goal in this project is to implement the nature of fuzzy classification into cork quality.

Acknowledgment

The authors would like to thank C. Campos from Corcho y Tecnologia S. L. and A. Sabate from Corchos de Merida S.A. for their technical support about practical aspects of cork classification.

References

J. Francisco Duque-Carrillo (M’86) received the M.Sc. degree in electronic physics from the University of Sevilla, Spain, and the Ph.D. degree from the University of Extremadura, Badajoz, Spain, in 1979 and 1984, respectively.

In 1986 and 1987, on a NATO Fellowship, he was a Visiting Scholar at the Electrical Engineering Department of Texas A&M University, College Station. In 1988, he was with AT&T Microelectronics in Madrid, Spain, and Allentown, PA. Currently, he is with the University of Extremadura, where he holds a Professor position. His research interests include analog and mixed-mode integrated circuit design for signal and neural information processing.

Edgar Sánchez-Sinencio (S’72–M’74–SM’83–F’92) received the M.S.E.E. degree from Stanford University, CA, and the Ph.D. degree from the University of Illinois at Urbana-Champaign, in 1970 and 1973, respectively. He did industrial postdoctoral work with Nippon Electric Company, Kawasaki, Japan, in 1973 and 1974.

Currently, he is with the Department of Electrical Engineering, Texas A&M University, College Station, as a Professor. He is the coauthor of Switched-Capacitor Circuits (New York: Van Nostrand-Reinhold, 1984) and coeditor of Artificial Neural Networks: Paradigms, Applications, and Hardware Implementations (New York: IEEE Press, 1992). His research interests include solid-state signal processing circuits, including BiCMOS, CMOS neural networks, and fuzzy and wavelets implementations.

Dr. Sanchez-Sinencio has been the Guest Editor or Coeditor of three special issues on neural-network hardware (IEEE TRANSACTIONS ON NEURAL NETWORKS, March 1991, May 1992, and May 1995) and one special issue on low-voltage low-power analog and mixed-signal circuits and systems (IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, November 1995). He was the IEEE/CAS Technical Committee Chairman on Analog Signal Processing from 1994 to 1995. He has been Associate Editor for different IEEE magazines and transactions since 1982. He was the IEEE Video Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS. He was also the IEEE Neural Network Council Fellow Committee Chairman in 1994 and 1995. He was a member of IEEE CAS Board of Governors (1990–1992). He was the 1993–1994 IEEE Circuits and Systems Vice President-Publications and a member of the IEEE Press Editorial Board. In 1995, he received an Honoris Causa Doctorate from the National Institute for Astrophysics, Optics and Electronics, Puebla, Mexico. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II (1997–1999).