

Automatic Classification of Fruit Defects based on Co-Occurrence Matrix and Neural Networks

Giacomo Capizzi*, Grazia Lo Sciuto[†], Christian Napoli[‡], Emiliano Tramontana[‡], Marcin Woźniak[§]

*Department of Electrical and Informatics Engineering, University of Catania, Viale A. Doria 6, 95125 Catania, Italy
Email: capizzi@dieei.unict.it

[†]Department of Electronic Engineering, University of Roma Tre, Via della Vasca Navale 84, 00146 Roma, Italy
Email: glosciuto@dii.unict.it

[‡]Department of Mathematics and Informatics, University of Catania, Viale A. Doria 6, 95125 Catania, Italy
Email: napoli@dmi.unict.it, tramontana@dmi.unict.it

[§]Institute of Mathematics, Silesian University of Technology, Kaszubska 23, 44-100 Gliwice, Poland
Email: marcin.wozniak@polsl.pl

Abstract—Nowadays the effective and fast detection of fruit defects is one of the main concerns for fruit selling companies. This paper presents a new approach that classifies fruit surface defects in color and texture using Radial Basis Probabilistic Neural Networks (RBPNN). The texture and gray features of defect area are extracted by computing a gray level co-occurrence matrix and then defect areas are classified by the applied RBPNN solution.

Keywords—Co-occurrence Matrix, Texture Analysis, Pattern Recognition, Probabilistic Neural Network.

I. INTRODUCTION

IN ORDER to ensure the quality standard required in orange production lines, companies need trained people to inspect the fruits while they move in a conveyor belt. These experts classify oranges according to several categories based on visual features. However, for fast and precise quality standards, this approach is not competitive and sometimes unreliable. Therefore is paramount to use of an automatic system based on intelligent methods [1], [2], [3], [4], [5], [6]. The aim of this paper is to investigate the applicability of the Artificial Intelligence (AI) methods based on soft computing for detection of external defects of the fruits. Several methods based on more specific image acquisition are reported in [7], [8], [9], [10], [11], [12], [13]. Wen and Tao developed a near-infrared vision system for automatic apple defect inspection, see [14]. While some recent advances in feature extraction for images and biometrics are reported in [15], [16], [17], [18], [19]. Zion et al. introduced a computerised method to detect the bruises of Jonathan, Golden Delicious, and Hermon apples from magnetic resonance images by threshold technique. The algorithm was only able to discriminate between all-bruised and non-bruised apples and was not applicable to on-line detection. Pla and Juste presented a thinning algorithm to discriminate between stem and body of the apples on monochromatic images. However, the task of classifying the calyx and defected parts in real-time was missing. Yang and Marchant used the ‘flooding’ algorithm for initial segmentation and ‘snakes’ algorithm for refining the boundary of the blemishes on the monochromatic images of apples. Miller

et al. in [20] compared different neural network models for detection of blemishes of various kinds of apples by their reflectance characteristics.

From the said research, and according to the current literature, we can conclude that Multi-Layer Back Propagation (MLBP) gives proper recognition rates and also that increased complexity of the neural network system did not yield to better results [21]. Leemans, segmented defects of ‘Golden Delicious’ apples by a pixel-wise comparison method between the chromatic (RGB) values of the related pixel and the color reference model. The local and global approaches of comparison were effective, but more research was needed. In his further research work, Leemans [22] used a Bayesian classification method for pixel-wise segmentation on chromatic images of ‘Jonagold’ apples. Machine vision systems are successfully used for recognition of greenhouse cucumber fruit using computer vision [23]. A method for the classification and gradation of different grains (for a single grain kernel) such as groundnut, Bengal gram, Wheat etc., is described in [24]. The effect of foreign bodies on recognition and classification of food grains is given in [25]. Some researchers have used a neural network approach to the color grading of apples [26]. All these studies show that it is hard to define geometric or spectral properties for the fruit skin.

Most of the industry treatments such as washing and packing are highly automated, while the most important verification steps (e.g., inspection and grading in quality) are still performed manually, around the world. For this reason, we propose an efficient classifier based on fruit skin texture analysis, which can be implemented in manufacturing expert systems [27]. A new automatic classifier of oranges defects based on co-occurrence matrix and probabilistic neural networks is presented in the following sections. Orange is an important fruit of Mediterranean countries and it is well-known for its considerable anti-microbial, anti-viral, potent anti-oxidant properties. High-quality products are the basis for the success in today’s highly competitive market. Currently, manual inspection is being used in order to determine the orange quality. The increasing demand for quality assurance

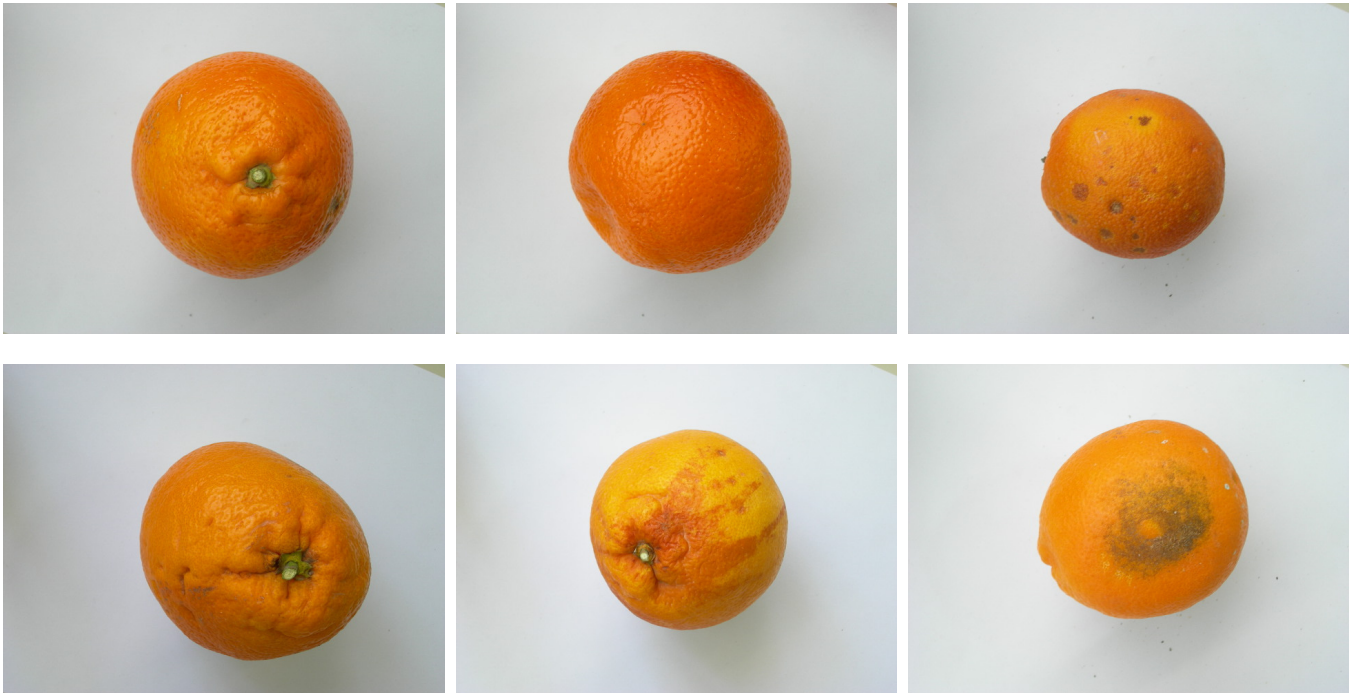


Fig. 1. Different sample of oranges used for this work. From left to right and top to bottom: Two normal oranges, an orange with several surface defects, a morphology defect, an orange with color defect, and an orange affected by a black mould.

requires simple and reliable sorting methods. The use of computer vision systems enables the detection of external quality defects. The objective of this study was to investigate the applicability of a method for detection of external defects and an automatic orange's classification system. For this purpose, the development of a classifier based on textural features of images, captured with a digital camera, was evaluated. The proposed technique is very robust and can identify specific defects like: surface defect, morphological defect, color defect, black mould or recognise a normal fruit. In fact, our RBPN model is able to correctly attribute the samples to the correct defect groups with an overall error of 2.75%.

II. FRUIT CLASSIFICATION TECHNIQUE

In previous years, several types of image analysis techniques are applied to analyse the agricultural images such as fruits and vegetables, for recognition and classification purposes. The most popular analysis techniques that have been used for both recognition and classifications of two dimensional (2D) fruit images are color-based and texture-based analysis methods.

A. Fruit classification based on shape

Shape based classification of fruits is based on various features like area, perimeter, major axis length and minor axis length. For calculating shape features an RGB image is converted into a gray scale image. After conversion into gray scale, the image represents a luminance intensity scale. There is a difference in intensity values for an object to be classified and its background, hence a threshold value is used to separate an object from its background. According to this

threshold value, a gray scale image is converted into a binary image in which the value greater than the threshold is 1 and the value lower than the threshold is 0. With the help of this binary image different shape features are computed. The most common shape features are computed from the image area, perimeter, major axis length and minor axis length.

B. Fruit classification based on color

RGB color space is converted into another color space such as HSV and for all the converted color space values, the mean and standard deviation are calculated. Each fruit image gives different values of mean and standard deviation, therefore assisting its classification.

1) *HSV Color Space*: HSI stands for hue, saturation and intensity. Then, for an image the color attribute is given by hue and the amount by which the pure color is diluted by white is given by saturation. The RGB components are separated from the original image, and the Hue (H), Saturation (S) and Intensity (I) components are extracted from RGB components. Equations (1), (2) and (3) are used to evaluate Hue, Saturation and Intensity of the image samples.

$$H = \begin{cases} \theta & B \leq G \\ 360 - \theta & B \geq G \end{cases} \quad (1)$$

$$\theta = \cos^{-1} \left\{ \frac{1}{2} \left(\frac{|(R-G)+(R-B)|}{[(R-G)^2+(R-B)\sqrt{G-B}]} \right) \right\}$$

The saturation component is given by:

$$S = 1 - \left(\frac{3}{R+G+B} \right) [\min(R, G, B)] \quad (2)$$



Fig. 2. An orange with some surface defects and the segmentation preprocessing result: the background is removed in order to analyse the orange surface only.

The intensity component is given by:

$$I = \frac{1}{3}(R + G + B) \quad (3)$$

C. Fruit classification based on texture

Texture is classified by the spatial distribution of gray levels in a neighborhood. It also helps in surface and shape determination. Gray level co-occurrence matrix is used to calculate different texture features [28]. There are two methods that can be used to calculate the texture feature of an image. One is the statistical texture analysis; the other is the structure of texture analysis. The former is the most conventional. Statistical texture analysis methods include spatial autocorrelation method, Fourier power spectrum method, Co-occurrence matrix method, gray level difference statistics method and trip length statistics method. Color mapping co-occurrence matrix (CMCM) is used to extract the texture information from a skin image. Gray level co-occurrence matrix (GLCM) is used to extract texture features in an image. It represents the form of tabulation which contains different combinations of pixel brightness values (gray levels) that occurs in an image. To calculate the different texture features like entropy, energy, homogeneity and dissimilarity, a gray level co-occurrence matrix is created.

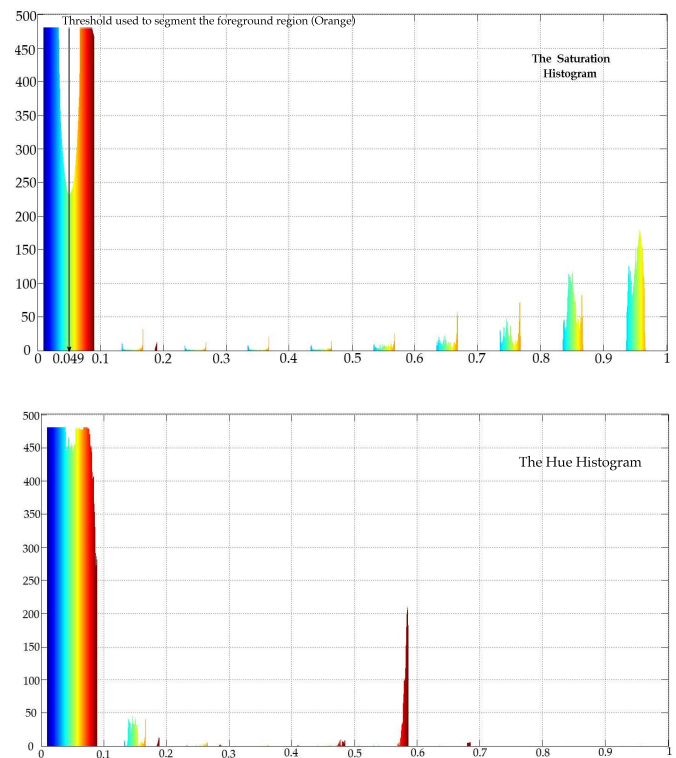


Fig. 3. The saturation (above) and hue (below) histogram related to the orange depicted in Fig. 2. The result shown in Fig. 2 is obtained by thresholding the saturation histogram shown here.

III. CLASSIFICATION PROCESS OF THE ORANGE DEFECTS

The European Union defines three quality classes (*extra*, *class I* and *class II*) for the fresh oranges with the tolerances of 5% and 10% by number or weight, respectively. The oranges in the *extra* class must be of superior quality with no defects or irregularity in shape, whereas the *class I* and *class II* can contain defects up to 1 cm² and 2.5 cm², respectively. All defect types in oranges contribute roughly equally to the final grading decision as local color, structural or textural variation of oranges.

In our classifier the defect detector uses a set of masks to recognise regions on the orange image. The defect is characterised by a discontinuity in the skin pigmentation. The features extracted from the orange images in either spatial or frequency domain can be used for classification. The external surface quality is directly related to the marketing and sales. Our automatic grading system can significantly improve the accuracy and consistency, while at the same time eliminating the subjectivity of manual inspection. The defects that we are able to detect on the external surface are caused by two reasons:

- Pre-harvest and Post-harvest diseases, like: diplodia and phomopsis stemend rot, splitting, pitting, green and blue mold, sour and brown rots, anthracnose, etc.
- Mechanical damages during transportation.

The defects on the orange fruit are characterised by different

textures. Among various textures, we categorised defect types like:

- Pitting, which is caused by mechanical damage or reduced gas exchange during transportation. Pits can coalesce to form irregular patches and brown to black blemishes.
- Splitting is caused by the inability of the outer skin to hold the weight of the whole fruit. The outer skin of the citrus fruit splits and the inner pulp gets exposed. The defective region is usually brighter when compared with the normal skin.

For the research we used a data set obtained after processing high quality orange images and defective quality orange images, for a total of 400 acquired images that are obtained by rotating and rearranging fruit samples. Color values (RGB-channels) as local features, are directly related with the images, so in our classifier they were introduced into the system without any change.

IV. PROPOSED TECHNIQUE FOR ORANGE FEATURE EXTRACTION

The foreground region (orange) then can be segmented out from the background by using the Hue and Saturation histograms of the image. In order to extract the features, the pictures have been firstly segmented and the background removed. This latter stage is very fast and its computational weight can be neglected, since it only uses the saturation histogram and a saturation threshold having value 0.049. Figure 2 shows an orange with several surface defects and the segmentation result, while Figure 3 shows the saturation and hue histogram of the same orange.

In addition to the segmentation using its HSV representation, we use color histogram and GLCM (Gray level co-occurrence matrix) to find quality classes corresponding to the orange qualities. For practical applications, the original input variables are typically preprocessed to transform them into some new space of variables where the pattern recognition problem will be easier to solve. This preprocessing stage is called *feature extraction*, see [29]. Preprocessing is performed in order to speed up computation and improve the classification performance [30]. We then selected useful features that are fast to compute and allow easy discrimination [31]. A feature set suitable for the classification should be insensitive to significant translations and have a very low correlation. Therefore, we base the extraction of features on the array of co-occurrence matrix, see [32]. Co-occurrence matrix is a single level dependence matrix that contains relative frequencies of two coordinate elements separated by a distance d . As you move from one pixel to another on the image, entries of the initial and final pixels become the coordinates of the co-occurrence matrix to be incremented, which in the end will represent structural characteristics of the image. Therefore, moving in different directions and distances on the image will lead to different co-occurrence matrices.

For each image we have calculated three co-occurrence matrices: one for each channel Hue, Saturation, Value. The ob-

tained features represent the area of the orange, the background (the contrast, like gray level uniformity), gray level correlation between neighbours, sum average and sum variance. The features were calculated from the normalised co-occurrence matrix for $d = 1$ and four main directions: 0° , 45° , 90° and 135° .

Since the use of the co-occurrence matrices leads to a course of dimensionality, we use five statistical descriptors of these matrices which use a co-occurrences gray level matrix. This is to calculate the number of adjacent pixels repetitions with the same gray level in the whole image. The statistical descriptors are the five textural features derived from the co-occurrence matrices.

Angular Second Moment: Consists of the sum of the squared elements in the co-occurrences matrix taken by pairs

$$ASM = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} [P_1(i, j)]^2. \quad (4)$$

Contrast: The global contrast of the image (also known as variance or *inertia*) measures the contrast intensity between a pixel and its neighbours

$$Contrast = \frac{1}{(N_g - 1)^2} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - j)^2 P_1(i, j). \quad (5)$$

Correlation: Measures the relation of a pixel and its neighbours. The degree in which if the gray level of a pixel increases, its neighbour also increases

$$Correlation = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (1 - \mu_x)(j - \mu_y) P_1(i, j)}{\sigma_x \sigma_y}, \quad (6)$$

where the expected values are expressed as

$$\begin{aligned} \mu_x &= \sum_{i=1}^{N_g} i \sum_{j=1}^{N_g} P_1(i, j) \\ \mu_y &= \sum_{j=1}^{N_g} j \sum_{i=1}^{N_g} P_1(i, j) \end{aligned} \quad (7)$$

and σ_x , σ_y which are the variances of all the values of the class that feature belongs to

$$\begin{aligned} \sigma_x^2 &= \sum_{i=1}^{N_g} (i - \mu_x)^2 \sum_{j=1}^{N_g} P_1(i, j) \\ \sigma_y^2 &= \sum_{j=1}^{N_g} (j - \mu_y)^2 \sum_{i=1}^{N_g} P_1(i, j) \end{aligned} \quad (8)$$

Gradient Module: Is a measure of the degree of asymmetry of a distribution around the mean. It is obtained by calculating the third central moment of the distribution. If the obtained value is zero, it means that it is centered (like the normal distribution). If it is positive, it is asymmetrical to the right, and if it is negative, to the left

$$GM = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_1(i, j) f(i)^2 \quad (9)$$

$$f(i) = i - N_g + 1$$

Intensity symmetry: Is a measure of the variation of the texture if the grey levels are reversed

$$IS = 1 - \sum_{i,j=1}^{N_g} |P_1(i, j) - P_1(N_g - 1 - i, N_g - 1 - j)|. \quad (10)$$

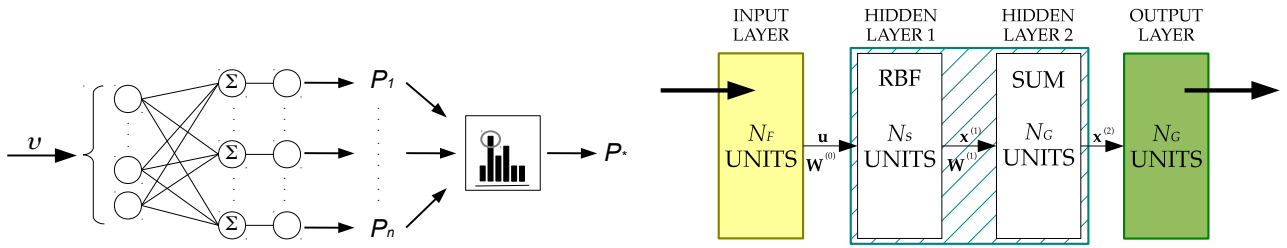


Fig. 4. A representation of our Radial Basis Probabilistic Neural Network (RBPNN) with maximum probability selector module. On the right the RBPNN layers model: N_F is the number of features, N_S is the number of samples and N_G is the desired number defects to identify.

Firstly, we use the information from the co-occurrence matrices [33]. These matrices represent the spatial distribution and the dependence of the gray levels within a local area. Each (i, j) entry of the matrices represents the probability of going from one pixel with a gray level (i) to another with a gray level (j) under a predefined distance and angle. More matrices are formed for specific spatial distances and predefined angles. From these matrices, sets of statistical measures are computed (called feature vectors) for building different texture models. In our case four angles, namely 0° , 45° , 90° , and 135° , are considered as well as a predefined distance of one pixel, in the formation of the co-occurrence matrices. Therefore, we have formed four co-occurrence matrices. A contrast equation (5) is used to calculate five parameters respectively, 0° , 45° , 90° , 135° , at limited distance one and 0° at a limited distance ten. This feature set extraction has a low dimensionality and a good discriminatory power. The features, e.g., mean and standard deviation, are extracted from each of the detail as well as the approximation sub-windows, and then fed into the Artificial Neural Network (ANN) classifier for defect classification. If the edges are not clearly defined, then the features extracted from such images are not useful. Training, testing, and validation of neural networks are performed using sample images.

V. THE SELECTED RBPNN

The proposed RBPNN (Fig. 4) is an implementation of a statistical algorithm called kernel discriminant analysis, see [34], in which the operations are organized into a multi-layered feed-forward network with four layers: an input layer, a pattern layer (the first hidden layer), a summation layer (the second hidden layer), and an output layer. Basically a RBPNN consists of an input layer, which represents the input pattern or feature vector. The input layer is fully interconnected with the hidden layer, which consists of the example vectors (the training set for the PNN [35], [36]). One other important element of the RBPNN is the output layer and the determination of the class for which the input layer fits. This is done through a winner-takes-all approach [37]. The output class node with the largest activation represents the winning class. While the class nodes are connected only to the example hidden nodes for their class, the input feature vector connects to all examples and therefore influences their activations. It is therefore the sum of the example vector activations that determines the class

of the input feature vector. In RBPNN algorithm, calculating the class-node activations is a simple process. For each class node, the example vector activations are summed, which are the sum of the products of the example vector and the input vector, see [38]. Input neurons are used as distribution units that supply the same input values to all the neurons in the first hidden layer (such neurons are called *pattern units*). Each pattern unit performs the dot product (\cdot) of the input pattern vector \mathbf{u} by a weight vector $\mathbf{W}^{(0)}$. Then each pattern unit performs a nonlinear operation on the result. This nonlinear operation gives output $\mathbf{x}^{(1)}$ that is handed to the following summation layer. While the common sigmoid function is used for a standard FFNN, in our BPTA for presented PNN the activation function is exponential. Therefore for the j neuron the output is where σ represents the statistical distribution spread

$$\mathbf{x}_j^{(1)} \propto \exp\left(\frac{\|\mathbf{W}^{(0)} \cdot \mathbf{u}\|}{2\sigma^2}\right). \quad (11)$$

In the proposed model, while preserving the PNN topology [39], in order to obtain RBPNN capabilities, the activation function has been substituted with a Radial Basis Function (RBF). Moreover, there is the equivalence between the $\mathbf{W}^{(0)}$ vector of weights and the centroids vector of RBNN, which in our classifier are computed as the statistical centroids of all the given input sets. We name ρ the chosen RBF, then the new output of the first hidden layer for the j neuron is

$$\mathbf{x}_j^{(1)} \triangleq \rho\left(\frac{\|\mathbf{u} - \mathbf{W}^{(0)}\|}{\beta}\right), \quad (12)$$

where β is the distribution shape control parameter, similar to σ used in (11).

The second hidden layer in our RBPNN is a PNN. It computes weighted sums of received values from the preceding neurons. This second hidden layer is called summation layer with the output of the k summation unit

$$\mathbf{x}_k^{(2)} = \sum_j \mathbf{W}_{jk} \mathbf{x}_j^{(1)}, \quad (13)$$

where \mathbf{W}_{jk} represents the weight matrix. Such weight matrix consists of a weight value for each connection from the j pattern unit to the k summation unit. The summation units work as the neurons of a linear perceptron network. The training for the output in the applied model is performed similarly to RBNNs.

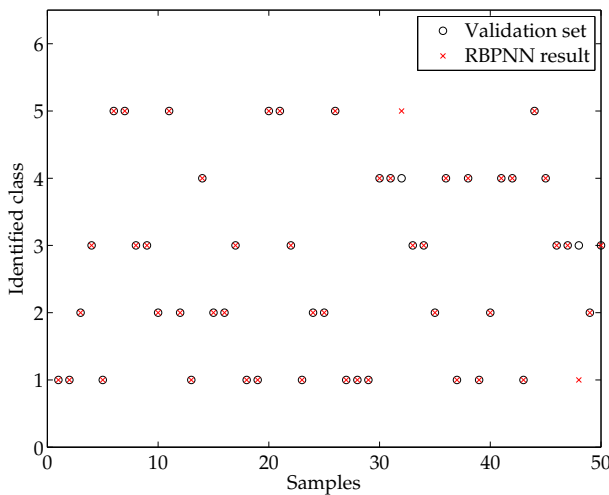


Fig. 5. The results of the identification proposed by our system at validation time: the circles show the effective class of the oranges, while the crosses show the identification output given by the implemented system.

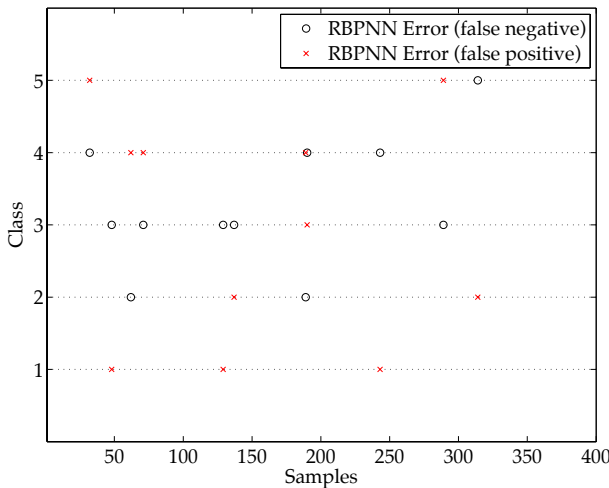


Fig. 6. The errors of identification of the proposed system: the circles identify the false negative results, while the crosses the false positive results.

VI. EXPERIMENTAL RESULTS AND CONCLUSIONS

In our RBPNN classifier, defects were grouped into categories: surface defect as bruises (class 2), morphological defects (class 3), slight color defects (class 4), black mould (class 5) and a category of good fruit (class 1). The proposed classifier has been tested on a large number of image orange samples collected in a database. This section shows 400 samples of different defects of fresh orange surface include stab wounds, bruise, abrasion, sunburn, injury, hail damage, cracks and insect pest damage and good oranges. Fig. 6 shows the different recognitions. Our RBPNN model is able to correctly attribute the orange samples to the correct defect groups with an overall error of 2.75% (see Table I).

The proposed solution enabled us to obtain a fast and efficient automatic classification system for fruits defects.

TABLE I
A BRIEF STATISTICAL ANALYSIS OF THE RESULTS BY CLASS.

Class	Fruit condition	Number of samples	False negative	False positive
1	Normal fruit	85	0	3
2	Surface defect	97	2	2
3	Morphological defect	71	5	1
4	Color defect	83	3	3
5	Black mould	64	1	2
Overall classification		400	389	2.75%

Although in an early stage of development, still such a system could be proposed on large scale for industrial applications, as well as for an implementation into programmable hardware.

REFERENCES

- [1] M. S. Pukish, P. Rózycki, and B. M. Wilamowski, "Polynet: A polynomial-based learning machine for universal approximation," *IEEE Trans. Industrial Informatics*, vol. 11, no. 3, pp. 708–716, 2015. [Online]. Available: <http://dx.doi.org/10.1109/TII.2015.2426012>
- [2] P. D. Reiner and B. M. Wilamowski, "Efficient incremental construction of RBF networks using quasi-gradient method," *Neurocomputing*, vol. 150, pp. 349–356, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0925231214012946>
- [3] A. Horzyk, "How does generalization and creativity come into being in neural associative systems and how does it form human-like knowledge?" *Neurocomputing*, vol. 144, pp. 238–257, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.neucom.2014.04.046>
- [4] M. Woźniak, W. M. Kempa, M. Gabryel, and R. K. Nowicki, "A finite-buffer queue with a single vacation policy: An analytical study with evolutionary positioning," *Applied Mathematics and Computer Science*, vol. 24, no. 4, pp. 887–900, 2014. [Online]. Available: <http://dx.doi.org/10.2478/amcs-2014-0065>
- [5] A. Horzyk, "Innovative types and abilities of neural networks based on associative mechanisms and a new associative model of neurons," in *Artificial Intelligence and Soft Computing - 14th International Conference, ICAISC 2015, Zakopane, Poland, June 14-18, 2015, Proceedings, Part I*, L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, Eds., 2015, pp. 26–38. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19324-3_3
- [6] M. Woźniak and D. Połap, "On some aspects of genetic and evolutionary methods for optimization purposes," *International Journal of Electronics and Telecommunications*, vol. 61, no. 1, pp. 7–16, 2015. [Online]. Available: <http://dx.doi.org/10.1515/eletel-2015-0001>
- [7] C. Napoli, G. Pappalardo, and E. Tramontana, "Improving files availability for bittorrent using a diffusion model," in *23rd IEEE International WETICE Conference*. IEEE, 2014, pp. 191–196. [Online]. Available: <http://dx.doi.org/10.1109/WETICE.2014.65>
- [8] G. Capizzi, F. Bonanno, and C. Napoli, "Recurrent neural network-based control strategy for battery energy storage in generation systems with intermittent renewable energy sources," in *IEEE international conference on clean electrical power (ICCEP)*. IEEE, 2011, pp. 336–340. [Online]. Available: <http://dx.doi.org/10.1109/ICCEP.2011.6036300>
- [9] F. Bonanno, G. Capizzi, S. Coco, C. Napoli, A. Laudani, and G. Lo Sciuto, "Optimal thicknesses determination in a multilayer structure to improve the spp efficiency for photovoltaic devices by an hybrid fem–cascade neural network based approach," in *International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*. IEEE, 2014, pp. 355–362. [Online]. Available: <http://dx.doi.org/10.1109/SPEEDAM.2014.6872103>
- [10] C. Napoli, G. Pappalardo, E. Tramontana, and G. Zappalà, "A cloud-distributed gpu architecture for pattern identification in segmented detectors big-data surveys," *The Computer Journal*, p. bxu147, 2014. [Online]. Available: <http://dx.doi.org/10.1093/comjnl/bxu147>
- [11] M. Woźniak, "Fitness function for evolutionary computation applied in dynamic object simulation and positioning," in *IEEE SSCI 2014: 2014 IEEE Symposium Series on Computational Intelligence - CIVTS*

- 2014: 2014 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems, Proceedings. 9-12 December, Orlando, Florida, USA: IEEE, 2014, pp. 108–114. [Online]. Available: <http://dx.doi.org/10.1109/CIVTS.2014.7009485>
- [12] S. Fidanova, M. Paprzycki, and O. Roeva, “Hybrid GA-ACO algorithm for a model parameters identification problem,” in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, Warsaw, Poland, September 7-10, 2014.*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2014, pp. 413–420. [Online]. Available: <http://dx.doi.org/10.15439/2014F373>
- [13] I. Lirkov, M. Paprzycki, M. Ganzha, S. Sedukhin, and P. Gepner, “Performance analysis of scalable algorithms for 3d linear transforms,” in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, Warsaw, Poland, September 7-10, 2014.*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2014, pp. 613–622. [Online]. Available: <http://dx.doi.org/10.15439/2014F374>
- [14] Z. Wen and Y. Tao, “Dual-camera nir/mir imaging for stem-end/calix identification in apple defect sorting,” *Transactions of the ASAE*, vol. 43, no. 2, pp. 449–452, 2000.
- [15] M. Woźniak and Z. Marszałek, “An idea to apply firefly algorithm in 2D images key-points search,” *Communications in Computer and Information Science - ICIST'2014*, vol. 465, pp. 312–323, 2014.
- [16] C. Napoli, G. Pappalardo, E. Tramontana, Z. Marszałek, D. Połap, and M. Woźniak, “Simplified firefly algorithm for 2d image key-points search,” in *2014 IEEE Symposium on Computational Intelligence for Human-like Intelligence*. IEEE, 2014, pp. 118–125. [Online]. Available: <http://dx.doi.org/10.1109/CIHLI.2014.7013395>
- [17] M. Woźniak and D. Połap, “Basic concept of cuckoo search algorithm for 2d images processing with some research results - an idea to apply cuckoo search algorithm in 2d images key-points search,” in *SIGMAP 2014 - Proceedings of the 11th International Conference on Signal Processing and Multimedia Applications, Vienna, Austria, 28-30 August, 2014*, M. S. Obaidat, A. Holzinger, and E. Cabello, Eds., 2014, pp. 157–164. [Online]. Available: <http://dx.doi.org/10.5220/0005015801570164>
- [18] M. Kubanek, D. Smorawa, and T. Holotyak, “Feature extraction of palm vein patterns based on two-dimensional density function,” in *Artificial Intelligence and Soft Computing - 14th International Conference, ICAISC 2015, Zakopane, Poland, June 14-28, 2015, Proceedings, Part II*, L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, Eds., 2015, pp. 101–111. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19369-4_10
- [19] M. Kubanek, D. Smorawa, and M. Kurkowski, “Using facial asymmetry properties and hidden markov models for biometric authentication in security systems,” in *Artificial Intelligence and Soft Computing - 13th International Conference, ICAISC 2014, Zakopane, Poland, June 1-5, 2014, Proceedings, Part II*, L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, Eds., 2014, pp. 627–638. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-07176-3_55
- [20] W. Miller, J. Throop, and B. Upchurch, “Pattern recognition models for spectral reflectance evaluation of apple blemishes,” *Postharvest Biology and Technology*, vol. 14, no. 1, pp. 11–20, 1998.
- [21] C. Napoli, G. Pappalardo, and E. Tramontana, “A hybrid neuro-wavelet predictor for qos control and stability,” in *AI*IA 2013: Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2013, vol. 9119, pp. 527–538. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-03524-6_45
- [22] V. Leemans, H. Magein, and M.-F. Destain, “Defects segmentation on golden delicious apples by using colour machine vision,” *Computers and Electronics in Agriculture*, vol. 20, no. 2, pp. 117–130, 1998.
- [23] L. Zhang, Q. Yang, Y. Xun, X. Chen, Y. Ren, T. Yuan, Y. Tan, and W. Li, “Recognition of greenhouse cucumber fruit using computer vision,” *New Zealand Journal of Agricultural Research*, vol. 50, no. 5, pp. 1293–1298, 2007.
- [24] D. G. Savakar and B. S. Anami, “Recognition and classification of food grains, fruits and flowers using machine vision,” *International Journal of Food Engineering*, vol. 5, no. 4, 2009.
- [25] B. Anami and D. Savakar, “Effect of foreign bodies on recognition and classification of bulk food grains image samples,” *J Appl Comput Sci*, vol. 6, no. 3, pp. 77–83, 2009.
- [26] K. Nakano, “Application of neural networks to the color grading of apples,” *Computers and electronics in agriculture*, vol. 18, no. 2, pp. 105–116, 1997.
- [27] G. Cwikała, A. Sekala, and M. Woźniak, “The expert system supporting design of the manufacturing information acquisition system (mias) for production management,” *Advanced Materials Research*, vol. 1036, pp. 852–857, 2014.
- [28] J. M. Keller, S. Chen, and R. M. Crownover, “Texture description and segmentation through fractal geometry,” *Computer Vision, Graphics, and Image Processing*, vol. 45, no. 2, pp. 150–166, 1989.
- [29] C. M. Bishop et al., *Pattern recognition and machine learning*. Springer New York, 2006, vol. 1.
- [30] M. Woźniak, D. Połap, M. Gabryel, R. Nowicki, C. Napoli, and E. Tramontana, “Can we process 2d images using artificial bee colony?” in *Artificial Intelligence and Soft Computing*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2015, vol. 9119, pp. 660–671. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19324-3_59
- [31] B. Nowak, R. Nowicki, M. Woźniak, and C. Napoli, “Multi-class nearest neighbour classifier for incomplete data handling,” in *Artificial Intelligence and Soft Computing*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2015, vol. 9119, pp. 469–480. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19324-3_42
- [32] V. Arvis et al., “Generalization of the cooccurrence matrix for colour images: application to colour texture classification,” *Image Analysis & Stereology*, vol. 23, no. 1, pp. 63–72, 2011.
- [33] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, “Textural features for image classification,” *Systems, Man and Cybernetics, IEEE Transactions on*, no. 6, pp. 610–621, 1973.
- [34] B. D. Ripley, *Pattern recognition and neural networks*. Cambridge university press, 1996.
- [35] C. Napoli, G. Pappalardo, E. Tramontana, R. Nowicki, J. Starczewski, and M. Woźniak, “Toward work groups classification based on probabilistic neural network approach,” in *Artificial Intelligence and Soft Computing*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2015, vol. 9119, pp. 79–89. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-19324-3_8
- [36] F. Bonanno, G. Capizzi, G. Lo Sciuto, C. Napoli, G. Pappalardo, and E. Tramontana, “A novel cloud-distributed toolbox for optimal energy dispatch management from renewables in igss by using wrnn predictors and gpu parallel solutions,” in *Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2014 International Symposium on*. IEEE, 2014, pp. 1077–1084. [Online]. Available: <http://dx.doi.org/10.1109/SPEEDAM.2014.6872127>
- [37] C. Napoli, G. Pappalardo, and E. Tramontana, “An agent-driven semantical identifier using radial basis neural networks and reinforcement learning,” in *Proceedings of the XV Workshop Dagli Oggetti agli Agenti*, vol. 1260. CEUR-WS, 2014.
- [38] M. M. Gupta, L. Jin, and N. Homma, *Static and dynamic neural networks*, Hoboken. NJ: Wiley-Interscience, 2003.
- [39] F. Bonanno, G. Capizzi, and C. Napoli, “Some remarks on the application of rnn and prnn for the charge-discharge simulation of advanced lithium-ions battery energy storage,” in *International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*. IEEE, 2012, pp. 941–945. [Online]. Available: <http://dx.doi.org/10.1109/SPEEDAM.2012.6264500>